



Univerza v Mariboru

Fakulteta za elektrotehniko,
računalništvo in informatiko

Senad Salkanović

Tiled in Java igra z uporabo LibGDX

Projektna naloga

Maribor, maj 2025

KAZALO VSEBINE

1	UVOD	1
2	TEORETSKI PREGLED PODROČJA	2
2.1	LibGDX kot ogrodje za razvoj iger	2
2.2	Tiled in TMX format.....	2
2.3	Obstoječe rešitve in primeri uporabe	2
3	METODOLOGIJA.....	4
4	REZULTATI.....	5
4.1	Nalaganje in prikaz zemljevida.....	5
4.2	Premikanje, zaznavanje trkov in sovražniki	6
4.3	Vmesnik in stanje igre	6
5	DISKUSIJA.....	7
6	ZAKLJUČEK	8
7	LITERATURA IN VIRI	9

KAZALO SLIK

SLIKA 1: PRIKAZAN ZEMLJEVID, USTVARJEN V TILED IN UPORABLJEN V LIBGDX PROJEKTU, KJER SO RAZLIČNI SLOJI LOČENO OZNAČENI: TLA, OVIRE, INTERAKTIVNI OBJEKTI.	3
SLIKA 2: PRIMER ZEMLJEVIDA IZ TILED, UPORABLJENEGA V LIBGDX PROJEKTU	3
SLIKA 3: PRIKAZ NALOŽENEGA ZEMLJEVIDA IZ TILED V LIBGDX OKOLJU	5
SLIKA 4: PRIKAZ DELOVANJA TRKOV Z STATIČNIMI OBJEKTI.....	6

SEZNAM UPORABLJENIH SIMBOLOV IN KRATIC

Kratika Razvezan pomen (prevod)

TMX	Tiled Map XML (format zemljevidov iz orodja Tiled)
FPS	Frames Per Second (število sličic na sekundo)
AI	Artificial Intelligence (umetna inteligenca)
ECS	Entity-Component-System (arhitektura entitet in komponent)
IDE	Integrated Development Environment (integrirano razvojno okolje)
API	Application Programming Interface (programski vmesnik)
UI	User Interface (uporabniški vmesnik)
XML	eXtensible Markup Language (razširljiv označevalni jezik)
OOP	Object-Oriented Programming (objektno usmerjeno programiranje)
CPU	Central Processing Unit (centralna procesna enota)

Tiled in Java igra z uporabo LibGDX

Ključne besede: LibGDX, Tiled, 2D igra, TMX zemljevid, trki

Povzetek

V nalogi je predstavljena izdelava 2D igre z uporabo LibGDX in zemljevidov iz Tiled. Zemljevidi se nalagajo iz TMX datotek, igralec pa se premika po mapi in zaznava trke s statičnimi elementi.

Tiled and Java Game using LibGDX

Keywords: LibGDX, Tiled, 2D game, TMX map, collisions

Abstract

This paper presents the development of a 2D game using LibGDX and Tiled maps. Maps are loaded from TMX files, and the player moves through the world with basic collision detection.

1 UVOD

Razvoj računalniških iger je danes eno izmed tehnološko in kreativno najzahtevnejših področij računalništva. Igre niso več omejene na področje zabave, ampak so pomembne tudi v izobraževanju, simulaciji, promociji in terapevtskih aplikacijah. V tem kontekstu so se 2D igre ohranile kot učinkovita in dostopna oblika razvoja, saj omogočajo učenje osnov igralne logike, grafike in interaktivnosti brez pretirane kompleksnosti. Poleg samega programiranja iger je pomemben tudi vizualni vidik, zlasti izdelava in uporaba zemljevidov oziroma igralnega okolja. Eden najpogostejše uporabljenih orodij za ustvarjanje takšnih map je Tiled Map Editor, ki skupaj z ogrodjem LibGDX omogoča preprosto integracijo grafičnih podatkov v delujočo 2D igro.

V tem projektu sem raziskoval, kako s pomočjo LibGDX ogrodja in orodja Tiled izdelati funkcionalno 2D igro. Cilj je bil razviti ogrodje, ki omogoča dinamično nalaganje zemljevidov, prikaz ploščic, premikanje igralca in zaznavanje trkov. Raziskovalna vprašanja vključujejo: Kako LibGDX omogoča delo s TMX datotekami, ustvarjenimi v Tiled? Kako se s tem pristopom poenostavi prikaz zemljevidov in zaznavanje trkov? In ali lahko na osnovi teh orodij razvijemo razširljivo platformo za 2D igre?

V projektu sem izdelal osnovno 2D igro, v kateri se igralec premika po zemljevidu, zaznava trke z elementi okolice in interagira s sovražniki. Uporabil sem podporo, ki jo LibGDX nudi za delo z TMX datotekami, razredi za delo z zemljevidi, kamero, animacijami in teksturami. Implementiral sem osnovno logiko igre, igralno zanko, sistem kolizij ter prikaz igralnega stanja. Projekt je zasnovan modularno, z možnostjo razširitve z dodatnimi mapami, mehanikami in funkcijami.

Nadaljevanje naloge je razdeljeno na več poglavij. V drugem poglavju bom predstavil teoretične osnove uporabljenih tehnologij – LibGDX, Tiled in TMX format. V tretjem poglavju bom opisal uporabljena orodja, knjižnice in razvojne pristope. V četrtem poglavju sledijo rezultati izvedbe, medtem ko bom v petem poglavju razpravljal o uspešnosti rešitve, primerjal pristop z alternativami ter izpostavil možnosti izboljšav. Nalogo zaključim s povzetkom in predlogi za nadaljnji razvoj.

2 TEORETSKI PREGLED PODROČJA

V tem poglavju so predstavljene ključne tehnologije in koncepti, uporabljeni pri razvoju igre: ogrodje LibGDX, format zemljevidov TMX in orodje Tiled za ustvarjanje map. V nadaljevanju so opisane osnovne značilnosti LibGDX in njegovih komponent za prikazovanje zemljevidov ter metode za zaznavanje trkov. Poglavje je razdeljeno v tri podpoglavja, ki se osredotočajo na pregled ogrodja LibGDX, delo s Tiled zemljevidi ter obstoječe knjižnice in rešitve za upravljanje iger v tem kontekstu.

2.1 LibGDX kot ogrodje za razvoj iger

LibGDX je odprtokodno ogrodje za razvoj iger, napisano v Javi, ki omogoča večplatformni razvoj – igre lahko izvažamo za namizne operacijske sisteme, Android, iOS in splet (HTML5). Ogrodje vključuje module za grafiko (2D in 3D), zvok, fizične simulacije, delo z vhodnimi napravami ter podporo za tile-based zemljevide [1].

Ena izmed največjih prednosti LibGDX je vgrajen sistem za delo z zemljevidi, ustvarjenimi v orodju Tiled. Ključni razredi za to so TmxMapLoader za nalaganje TMX datotek, TiledMap kot predstavnik zemljevida, OrthogonalTiledMapRenderer za prikaz ploščic ter OrthographicCamera za upravljanje z vidnim poljem igre. LibGDX omogoča tudi integracijo fizikalnega sistema Box2D, ki se pogosto uporablja za zaznavanje trkov in simulacijo gibanja [2].

LibGDX vsebuje lasten "game loop" sistem z uporabo metode render(), ki se izvaja neprekinjeno, kar omogoča stalno posodabljanje logike igre in risanje na zaslon. V nasprotju s klasično Javo, kjer moramo sami ustvariti igralno zanko in upravljanje niti, LibGDX vse to integrira in optimizira za posamezne platforme [3].

2.2 Tiled in TMX format

Tiled je uveljavljeno odprtokodno orodje za ustvarjanje 2D zemljevidov. Uporabniki lahko z njim oblikujejo mape na osnovi ploščic (tiles), ki so lahko grafične slike poljubne velikosti. Mapa je sestavljena iz več slojev: vizualnih (tile layers), objektnih (object layers) in slojev s podatki (property layers). Tiled omogoča izvoz zemljevidov v TMX formatu, kar je XML-strukturirana tekstovna datoteka, ki vsebuje informacije o velikosti zemljevida, slojih, položaju ploščic, uporabljenih slikah in objektih [4].

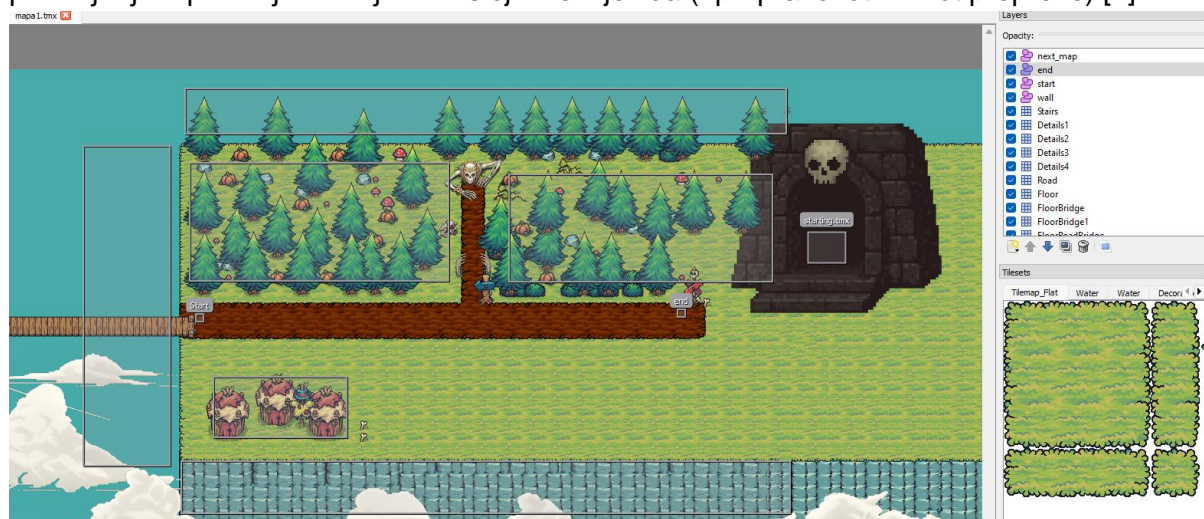
V TMX datoteki lahko vsak objekt nosi dodatne lastnosti (properties), kot so oznake za trke, začetne pozicije igralca, sovražnikov ali interaktivni elementi. LibGDX omogoča neposredno branje TMX datotek brez potrebe po ročnem razčlenjevanju XML vsebine. Ko zemljevid naložimo z TmxMapLoader, lahko dostopamo do njegovih slojev z metodami getLayers(), posamezne ploščice pa so na voljo prek TiledMapTileLayer.Cell objektov [5].

Zaradi podpore večslojnemu mapiranju lahko vizualne in logične podatke ločimo – npr. igralno okolje v enem sloju, trki v drugem, začetne pozicije likov v tretjem. Ta ločitev močno prispeva k čisti arhitekturi kode in hitrejši nadgradljivosti igre.

2.3 Obstoječe rešitve in primeri uporabe

LibGDX vključuje številne vnaprej pripravljene komponente, ki razvijalcem omogočajo hitro implementacijo osnovne funkcionalnosti igre. Na voljo je bogata dokumentacija, primeri (t.i. "demos") in odprtokodne igre, ki prikazujejo uporabo Tiled zemljevidov in zaznavanje trkov. Na primer, projekt **Super Koalio** v uradnem repozitoriju LibGDX demonstrira uporabo TMX map z razredom TiledMap, premikanje igralca in zaznavanje trkov s pomočjo Box2D [6].

Druge knjižnice, kot je **Ashley**, omogočajo uporabo "entity-component-system" (ECS) arhitekture znotraj LibGDX, kar olajša modularno sestavljanje igralnih elementov. Za zaznavanje trkov obstajajo pristopi z uporabo fizičnega motorja Box2D ali s preprostim preverjanjem položajev v objektnih slojih zemljevida (npr. pravokotniki kot prepreke) [7].



Slika 1: prikazan zemljevid, ustvarjen v Tiled in uporabljen v LibGDX projektu, kjer so različni sloji ločeno označeni: tla, ovire, interaktivni objekti.

Vir: Lastna slika



Slika 2: Primer zemljevida iz Tiled, uporabljenega v LibGDX projektu

Vir: Lastna slika

Takšna razmejitev slojev omogoča čisto ločevanje med prikazom in logiko igre. LibGDX podpira tudi animirane ploščice, dinamične spremembe zemljevida med izvajanjem igre in združevanje zemljevidov za ustvarjanje večjih igralnih svetov.

3 METODOLOGIJA

Raziskovalni problem, ki sem ga obravnaval v okviru tega projekta, se nanaša na integracijo zemljevidov, ustvarjenih z orodjem Tiled, v delujočo 2D igro, razvito z uporabo ogrodja LibGDX. Cilj naloge je bil razviti prototip igre, ki omogoča dinamično nalaganje TMX zemljevidov, premikanje igralca po teh zemljevidih, zaznavanje trkov z okolico ter osnovno interakcijo z elementi v svetu igre. Glavna hipoteza, ki sem jo želel preizkusiti, je bila: *LibGDX omogoča učinkovito in razširljivo implementacijo 2D igre s podporo za Tiled zemljevide, brez potrebe po dodatni obdelavi XML strukture.*

Pri razvoju sem uporabil več ključnih orodij in knjižnic. Ogradje **LibGDX** (različica 1.12.0) sem izbral zaradi njegove odlične podpore za delo s Tiled zemljevidi, večplatformske narave in dobre dokumentacije. Za razvojno okolje sem uporabil **IntelliJ IDEA Community Edition**, ki ponuja dobro podporo za Maven/Gradle projekte in LibGDX integracijo. Zemljevide sem ustvarjal v programu **Tiled**, ki omogoča vizualno urejanje in izvoz v TMX formatu. Za nalaganje in prikazovanje zemljevidov sem uporabil razreda `TmxMapLoader` in `OrthogonalTiledMapRenderer`, ki omogočata hitro integracijo zemljevida v igro. Za zaznavanje trkov sem uporabil `TiledMapTileLayer`, v kombinaciji s pravokotniki, označenimi v slojih tipa "object layer". V projektu nisem uporabil Box2D, saj sem želel ohraniti projekt preprost in osredotočen na ročno implementacijo logike trkov.

Razvoj sem začel z generiranjem osnovnega LibGDX projekta z uradnim zaganjalnikom (`gdx-setup.jar`), kjer sem določil podporo za namizje (desktop launcher) in uporabo standardne konfiguracije. Ustvaril sem osnovni razred igre (`Game`), kjer se ob zagonu naloži začetni zaslon (`Screen`). V razredu `GameScreen` sem inicializiral glavno kamero (`OrthographicCamera`), zemljevid (`TiledMap`) in renderer. S pomočjo razreda `MapLoader` sem zemljevid naložil iz TMX datoteke, ki je vsebovala vizualne ploščice in sloj z neprehodnimi objekti.

V nadaljevanju sem implementiral razred `Player`, ki vsebuje podatke o položaju, hitrosti in metodi `update()`, ki omogoča premikanje igralca z uporabo `Input` razreda. Za zaznavanje trkov sem iz sloja "object layer" prebral vse pravokotnike (`RectangleMapObject`) in jih ob vsakem premiku igralca preveril glede na trenutno pozicijo. Če se je igralec zaletel v neprehodni objekt, se je premik blokiral. Podobno sem kasneje dodal osnovne sovražnike, ki se gibljejo proti igralcu in sprožijo stanje poraza ob trku.

Igro sem testiral sproti, z izvajanjem aplikacije na namizni platformi. Testiral sem delovanje nalaganja različnih map, zaznavanje trkov na različnih pozicijah in delovanje kamere (scrolling). Ker LibGDX omogoča risanje le vidnega dela zemljevida, sem preverjal tudi delovanje optimizacij in gladkost prikaza pri večjih mapah. Meritve hitrosti izvajanja niso bile formalno izvedene, vendar sem opazoval FPS prikaz s pomočjo metode `Gdx.graphics.getFramesPerSecond()`.

V projektu sem upošteval več omejitev. Prva je bila odločitev, da ne vključim fizikalnega motorja Box2D, saj bi to povečalo kompleksnost projekta. Prav tako sem omejil uporabo animacij in se osredotočil na statične ploščice. Interakcija z elementi na mapi (npr. predmeti, vrata) ni bila implementirana, saj je cilj naloge bil prikaz osnovne funkcionalnosti – nalaganje zemljevida, premikanje in trki. Predpostavil sem, da je TMX datoteka pravilno strukturirana, z imeni slojev kot so `collision` ali `spawn`, ter da se velikost ploščic v celotni igri ne spreminja.

4 REZULTATI

V tem poglavju predstavljam končni rezultat izvedbe igre, ki vključuje prikaz zemljevida, premikanje igralca, zaznavanje trkov in osnovno logiko sovražnikov. Rezultati so podprti s slikami iz delujočega LibGDX okolja, kjer so vidni ključni elementi zasnove igre.

4.1 Nalaganje in prikaz zemljevida

Zemljevid je bil ustvarjen v programu Tiled in vsebuje več slojev: osnovni sloj za vizualni prikaz okolice, sloj za kolizije (object layer) in dodatne podatkovne sloje. V igri je zemljevid naložen s pomočjo razreda `TmxMapLoader`, ki ustvari objekt `TiledMap`. Za prikaz ploščic je uporabljen `OrthogonalTiledMapRenderer`, ki ob vsakem klicu metode `render()` nariše vse vidne sloje na zaslon.

Za premikanje igralčeve kamere sem uporabil `OrthographicCamera`, ki se premika skupaj z igralcem in s tem omogoča "scrolling" igre. Zemljevidi različnih velikosti se pravilno skalirajo glede na dimenzije kamere in resolucijo igre. Na sliki 1 je prikazan začetni zaslon igre s prikazanim zemljevidom.



Slika 3: Prikaz naloženega zemljevida iz Tiled v LibGDX okolju
Vir: Lastna slika

Igra podpira nalaganje več različnih zemljevidov – zamenjava TMX datoteke ne zahteva sprememb v kodi, kar omogoča modularno razširitev igre z novimi stopnjami. Zemljevidi se nalagajo iz mape `assets/maps`, kjer se samodejno prepoznajo razpoložljive TMX datoteke.

4.2 Premikanje, zaznavanje trkov

Igralec se premika po mapi z uporabo smernih tipk, hitrost premikanja je konstantna. Premikanje poteka v štirih smereh (gor, dol, levo, desno), koordinatni sistem pa temelji na enotah ploščic (npr. 32x32 točk). Igralec je predstavljen kot Sprite, ki se ob vsaki spremembi pozicije premakne in se nato preveri trk z neprehodnimi elementi.

Za zaznavanje trkov sem iz object layerja zemljevida pridobil vse `RectangleMapObject` objekte, ki jih LibGDX omogoča brati neposredno prek `MapLayer.getObjects()`. Ko se igralec premakne, preverim, ali njegov pravokotnik prekriva katero izmed ovirov. Če pride do prekrivanja, se gibanje razveljavi. Ta pristop omogoča natančno detekcijo in je dovolj hiter za igre z manjšim številom trdih objektov.

Poleg igralca sem dodal osnovne sovražnike, ki uporabljajo enostavno logiko sledenja igralcu. Ko sovražnik doseže igralca, se sproži stanje poraza in igra prikaže ustrezno obvestilo. V naslednjem GIF-u je prikaz delovanja trkov v igri



Slika 4: Prikaz delovanja trkov z statičnimi objekti

4.3 Vmesnik in stanje igre

Za prikaz uporabniškega vmesnika sem uporabil razred `BitmapFont` za prikaz osnovnih informacij, kot so koordinate igralca, FPS in trenutno stanje (igranje, poraz). Uporabniški vmesnik se prikazuje nad glavno igro z uporabo `SpriteBatch` objekta in ločene kamere, kar omogoča neodvisen prikaz elementov uporabniškega vmesnika od igralnega sveta.

Stanja igre so implementirana kot preklap med različnimi zasloni (`Screen`) – npr. začetni meni, igralni zaslon, konec igre. To omogoča boljšo organizacijo kode in ločevanje logike za posamezne faze igre.

5 DISKUSIJA

Rezultati projekta kažejo, da je LibGDX izredno primerno ogrodje za razvoj 2D iger z uporabo Tiled zemljevidov. Z uporabo razredov, kot so TmxMapLoader, OrthogonalTiledMapRenderer in MapObject, je bilo mogoče implementirati stabilno rešitev brez potrebe po dodatnih knjižnicah ali lastnem branju XML datotek. To potrjuje hipotezo, da LibGDX zagotavlja visoko stopnjo abstrakcije in optimizirano podporo za ključne funkcionalnosti v takšnih igrarh.

Nalaganje zemljevida se je izkazalo kot enostavno in zanesljivo. Modularnost TMX formata omogoča enostavno dodajanje novih vsebin, kar je pomembno za nadaljnjo razširljivost. Podpora za različne vrste slojev v Tiled omogoča čisto ločitev vizualnega in logičnega dela igre – kar je bil eden izmed ciljev naloge. Premikanje igralca in zaznavanje trkov sta delovala stabilno, brez zamikov ali napak, kar je potrdilo ustreznost izbranega pristopa (detekcija trkov s pravokotniki).

Omejitev projekta je predvsem v preprostosti implementacije – igra ne vključuje fizikalnega motorja (npr. Box2D), kar bi omogočilo bolj realistično interakcijo z okolico. Prav tako ni implementirane podpore za animacije, zvok, shranjevanje stanja ali večigralski način. Kljub temu pa so osnovne komponente igre postavljene na način, ki omogoča enostavno nadgradnjo.

V primerjavi z ročno implementacijo igralnega pogona v čisti Javi je bila uporaba LibGDX bistveno hitrejša in učinkovitejša. Dokumentacija in primeri uporabe, ki jih ponuja skupnost, so omogočili hitro odpravljanje težav. Uporaba LibGDX v povezavi s Tiled se je tako izkazala kot zelo primerna izbira za razvoj preprostih, modularnih 2D iger.

Projekt bi lahko v prihodnosti nadgradil z uporabo Box2D za fiziko, s podporo za skriptiranje dogodkov v Tiled (npr. Lua), z animiranimi ploščicami in dodajanjem sistema nalog ali nivojev. Tako bi bil sistem še bolj uporaben tudi za kompleksnejše igre ali prototipe.

6 ZAKLJUČEK

V okviru seminarske naloge sem uspešno razvil osnovno 2D igro, ki uporablja ogrodje LibGDX in zunanje zemljevide, ustvarjene v urejevalniku Tiled. Osrednji cilj naloge – vzpostaviti delujoče okolje, kjer se TMX zemljevidi samodejno nalagajo, prikazujejo in služijo kot osnova za igralno logiko – je bil dosežen. S tem sem dokazal, da LibGDX ponuja vgrajeno podporo za delo s Tiled, kar omogoča enostavno integracijo grafične predstavitve in logičnih slojev v razvojni proces igre.

V nalogi sem implementiral premikanje igralca, zaznavanje trkov z uporabo objektnih slojev, osnovno umetno inteligenco sovražnikov ter prikaz igralnega vmesnika. Zemljevidi se naložijo neposredno iz TMX datotek in jih je mogoče poljubno zamenjevati, kar omogoča modularno nadgradnjo igre brez sprememb v izvorni kodi.

Poleg funkcionalne izvedbe sem v nalogi predstavil teoretične osnove uporabljenih orodij in tehnologij, pri čemer sem se osredotočil na strukturo TMX datotek, sistem slojev in delovanje igralne zanke v LibGDX. S tem sem pridobil boljše razumevanje sodobnih principov razvoja iger in integracije podatkovnih virov znotraj grafičnega pogona.

Pri delu sem se omejil na uporabo osnovnih komponent ogrodja LibGDX, brez vključitve dodatnih sistemov, kot so Box2D, animacije ali obsežen sistem nalog. Kljub temu pa je projekt vzpostavil trdne temelje, ki omogočajo nadaljnje nadgrajevanje. V prihodnje bi bilo smiselno razširiti projekt z bolj naprednimi sovražniki, scenariji, podporo za animirane ploščice ter vključitvijo zvoka in stanja igre.

Naloga je pokazala, da je povezava med Tiled in LibGDX zelo učinkovita in omogoča hitro prototipiranje ter modularen razvoj iger. Ugotovitve so uporabne tako v izobraževalne namene kot tudi kot osnova za nadaljnje projekte ali samostojno raziskovanje področja razvoja iger.

7 LITERATURA IN VIRI

- [1] LibGDX Documentation, »Tiled maps«. Pridobljeno: 2. april 2025. [Na spletu]. Dostopno na: <https://libgdx.com/wiki/graphics/2d/tile-maps>
- [2] B. L. Taylor, »Beginning Java Game Development with LibGDX«, Apress, 2015.
- [3] A. Cook, »Game Development Using LibGDX«, Packt Publishing, 2014.
- [4] Tiled Map Editor, »Documentation«, v1.10.2. Pridobljeno: 3. april 2025. [Na spletu]. Dostopno na: <https://doc.mapeditor.org/>
- [5] GitHub – LibGDX Examples. »Super Koalio«. Pridobljeno: 5. april 2025. [Na spletu]. Dostopno na: <https://github.com/libgdx/libgdx-demo-super-koalio>
- [6] M. Buckland, »Programming Game AI by Example«, Jones & Bartlett Learning, 2004.