

# **WriteSmart**

Projektni rad

Fakultet tehničkih nauka Novi Sad

Smjer: Računarstvo i automatika

Predmet: Arhitektura računara

Projekat: Edukativna igra

Autor: Darko Lukić

Mentor: Lazar Stričević

**Maj 2014**

## 1. O igri

Moglo bi se reći da je ovo arkadna igrice sa elementima kviz igara. Smatram da igre ne treba da budu sredstvo za opuštanje nego zabavan način za sticanje znanja i novih načina razmišljanja. U skladu sa tim pokušao sam osmisлити igru u kojoj bi korisnici mogli provjeriti i osvježiti svoje znanje, a prvenstveno naučiti nešto novo.

Pri dnu ekrana nalazi se lik ("sprite") koji je u ovom slučaju olovka sa kapom. Svaki nekoliko sekundi na vrhu ekrana pojavljuje se novo pitanje i tri ponuđena odgovora. Odgovori se nalaze u boksevima i poslije određenog vremena počinju da padaju, a vaš zadatak je da "pokupite" pravi. Naravno nije sve tako jednostavno, određene sile nasumično djeluju na olovku što otežava da "pokupite" željeni odgovor. Ove sile djeluju u različitim pravcima i intenzitetima i nikako ne možete da predvidite kako će olovka reagovati.

Na početku potrebno je odabrati težinu, odnosno nivo. Teži nivo donosi veći broj poena po tačno odgovorenom pitanju, ali zato imate manje vremena da razmislite o odgovoru. Nivo se bira strelicima, gore, dole, a igra počinje kada pritisnete desnu ili lijevu strelicu. Za upravljanje koriste se strelice, lijevo, desno.

Igra se završava kada odaberete pogrešan odgovor ili "spadnete" sa sveske. U tom trenutku pojavljuje se rank lista i ukoliko ostvarite dovoljan broj bodova za TOP5 vaš rezultat će se naći u listi, naznačen. Tada se očekuje od korisnika da upiše svoje ime i pritisne [Enter] kako bi se rezultat sačuvao u datoteku.

## 2. Realizacija

Projekat je programiran u C/C++, a pored standardnih C/C++ biblioteka korišćene su i:

- SDL (Simple Directmedia Layer) <http://www.libsdl.org/>
- SDL\_ttf [https://www.libsdl.org/projects/SDL\\_ttf/](https://www.libsdl.org/projects/SDL_ttf/)
- SDL\_image [https://www.libsdl.org/projects/SDL\\_image/](https://www.libsdl.org/projects/SDL_image/)
- SDL\_gfx <http://sourceforge.net/projects/sdlgfx/>

U početku sam se dvoumio između SFMLa i SDLa i za SDL sam se odlučio zbog veće zajednice, podrške i velikog broja projekata koje su radile generacije Računarstva i automatike prije mene, a bazirali su se upravo na SDLu. Takođe sam se dvoumio između dvije verzije SDLa, SDL 1.2 i SDL 2.0. Na prijedlog mentora, Lazara Stričevića, i zbog bolje podrške za Linux odabrao sam SDL 1.2.

Pored standardne SDL biblioteke pojavila se potreba i za korišćenje SDL\_ttf, SDL\_image i SDL\_gfx. SDL\_ttf je zadužen za prikazivanje teksta u igrici, a SDL\_gfx mi je bio potreban za rotaciju lika. Alternativno rješenje za rotaciju lika je crtanje niza slika lika pod različitim uglovima, a slike bi se birale zavisno od potrebnog ugla lika. Ovo rješenje je zahtjevalo više rada i bilo bi manje fluidno tako da sam se ipak odlučio za SDL\_gfx.

## 3. Problemi

Moram napomenuti da mi je ovo prva igrice i da sam posljednji put sa OOP u C++ susreo u trećem

razredu srednje iz Programiranja. S obzirom na to ovaj projekat je bio odličan način da se podsjetim nekih stvari i naučim nešto novo. Ovo je prvi put da nešto korisno naučim na fakultetu, tačnije dok sam na fakultetu, jer ovaj projekat radim "kod kuće".

Organizacija koda i datoteka je bio prvi problem sa kojim sam se suočio. Vjerovatno zbog neiskustva u pravljenju ovakvih aplikacija, nepoznavanja SDLa i lošeg znanja OOPa u C++ bilo je mi je jako teško da napravim dobru strukturu. Iz tog razloga često sam dosta stvari mijenjao i počinjao iz početka. Sada, kada je projekat završen uviđam greške i imam mnogo jasniju sliku strukture koja je bila pogodna za ovakve aplikacije. Sigurno bih napravio klasu za rad sa tekstovima koja bi obavezno sadržala podrazumjevanje vrijednosti, veličinu, boju i naziv fonta, zatim funkciju za prelamanje teksta i destruktor koji bi oslobađao zauzetu memoriju.

Za grafiku nisam smio koristiti slike sa Interneta, pa sam odlučio da ih sam napravim. Prvo sam nacrtao na papiru kako bi neka slika trebala da izgleda, zatim fotografisao telefonom i tu fotografiju koristio kao "pozadinski layer" u GIMPu. Preko toga u novim layerima sa Paintbrushom crtao sam novu sliku. Ograničenja C/C++ i same SDL biblioteke su me dosta usporavali. C/C++ ne posjeduje standardnu biblioteku sa funkcijom koja konvertuje integer u string, pa sam je morao sam napisati. Takođe, SDL nema mogućnost prelamanja teksta, pa sam morao napisati i funkciju za to koja nije radila dovoljno dobro.

Realistično kretanje olovke sam pokušao da realizujem tako što bi se olovka u prvom frejmu pomjerila za  $x$  tačaka, a sljedećem za  $(x - y)$  tačaka. Na sličan način napravio sam i naginjanje olovke zavisno od djelovanja sila.

## 4. Kompajliranje

Za kompajliranje aplikacije na Linuxu potrebno je preuzeti biblioteke koje sam naveo iznad, podnaslov Realizacija. U Terminalu potrebno je otići u direktorijum gdje se nalazi aplikacija, zatim pozvati komandu **make**. Nakon ove naredbe kreirana je izvršna datoteka i moguće ju je pokrenuti iz Terminala naredbom **./WriteSmart**.

## 5. Fajlovi potrebni za pokretanje

Sve datoteke u direktorijumu data su obavezni:

- Questions.txt
- Results.txt
- LiberationSans-Regular.ttf
- bg.png
- sprite.png

**Questions.txt** je moguće i namijenjen je za mijenjanje sadržaja s obzirom da se tu nalaze pitanja i odgovori koji se prikazuju u igrici. Pitanja se formiraju tako što se u svakoj liniji nalazi poseban parametar:

*pitanje*

*prvi odgovor*

*drugi odgovor*

*treći odgovor*

*redni broj tačnog odgovora – 1*

U tekstualnoj datoteci **Results.txt** nalazi se TOP5 igrača sa najviše bodova.

## 6. Kontrole

**Lijeva i desna strelica** – odabir nivoa, pomjeranje lika lijevo, desno,

**[Enter]** – pokretanje igre, potvrda imena i ponovno pokretanje igre.

## 7. Autorska prava

Aplikacije je pod GPL v3 licencom, detaljne informacije nalaze se u datoteci **licence/gpl.txt**. U direktorijumu **licence** se nalaze još i licenca za font.

## 8. Screenshot

