

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

PROJEKT IZ EKSPERTNIH SUSTAVA

**Sustav za određivanje zločinaca u sklopu tekstualne
avanture**

Filip Milić

Voditelj: *Alan Jović*

Zagreb, Prosinac, 2020

Sadržaj

1. Uvod	1
2. Ekspertni sustav	2
2.1 Detektiv	2
2.2 Pravila za ponašanje likova	3
2.3 Pravila za praćenje stanja u igri.....	4
3. Tekstualna avantura	5
4. Implementacija	7
5. Instalacija i uporaba.....	9
6. Opažanja, nedostaci i zaključak	10
7. Literatura	11

1. Uvod

Glavna inspiracija za ovaj projekt bila je nedavna popularnost igre *Among Us* i općeniti rast tzv. *social deduction* igara – u grupi ljudi postoji osoba koja je počinila zločin, i cilj ih je uhvatiti. U našem slučaju, konstruirali smo **ekspertni sustav koji igra upravo ulogu „detektiva“ – na temelju činjenica o zločinu i skupu pravila, sustav mora odrediti tko je sve mogući počinitelj.**

Složenost samih pravila se može kretati od relativno jednostavnih (čiji su otisci prstiju (ako ikakvi) su nađeni na oružju?) do složenih (na temelju vremena zločina i zadnjim poznatim položajima ljudi u kući, odrediti tko je uopće mogao na vrijeme doći do mjesta zločina i biti počinitelj – odnosno, ispitivanje alibija).

Međutim, ekspertni sustavi trebaju neku okolinu naspram koju dobivaju te činjenice. Sastavljanje pravila za detektiva je zapravo samo jedan dio konkretnog projekta. Drugi dio – i, kako se pokazalo, izazovniji – je osmišljanje upravo te okoline iz koje detektiv prima činjenice. Jedan od mogućih načina bio je, primjerice, dati korisniku da jednostavno ručno unese sve potrebne informacije.

Ali jedna od glavnih karakteristika prijašnje spomenutih inspiracija je njihova nepredvidljivost. Sustav bi se neporecivo zanimljivije ponašao ako bi okolina bila dinamična, barem nekako nalik na stvarnom svijetu iz koje bi se crpile činjenice. Druga mogućnost je, stoga, bila slučajno generiranje podataka. Međutim, slučajna generacija podataka bi sama po sebi zahtijevala još jedan, znatno kompliciraniji ekspertni sustav, koji bi generirao smislene zločine na temelju razumnih pravila.

Na kraju se odlučilo za spoj te dvije ideje – dopustiti korisniku interakciju, ali isto tako napraviti da su rezultati uvijek nepredvidljivi.

Rješenje je bilo u obliku **tekstualne avanture**. Korisnik – igrač – bi se stavio u ulogu počinitelja, i pokušao izvesti zločin. Nakon što se zločin izvede, detektiv – ekspertni sustav – bi pokušao zaključiti tko je počinitelj i „uhvatiti“ igrača.

U sklopu tekstualne avanture, igrač se nalazi u velikoj kući. Njegov cilj je ubiti danu žrtvu, tako da nisu uhvaćeni ni u djelu ni nakon što „detektiv“ dođe na mjesto zločina. Igrač se slobodno može kretati po kući. U kući se također nalazi više ljudi, i kako se igrač kreće, tako se i oni – što uključuje i žrtvu. Igrač skuplja objekte, moguća oružja, pokušava uspostaviti vjerodostojan alibi, te ostaviti što manje fizičkih dokaza.

2. Ekspertni sustav

Sam ekspertni sustav implementiran je kroz pravila u Prologu. Primjer pravila prikazan je na slici 1.

```
% Generally speaking, a culprit is someone who has means, motive and opportunity.
% However, in crime fiction (and in real life), motive itself does not necessarily
% carry much weight in the deduction process.
% Therefore, we define the culprit with someone who has the means at the opportunity.
culprit(X) :- person(X), opportunity(X), means(X).

% Alternatively, a person is immediately caught if there exists a witness to their crime.
% But since we directly only check witness(X, T) in the code, there's no need to
% include it in the ruleset here.
% culprit(X) :- person(X), person(Y), time(T), witness(Y, T), not(X = Y).

% Someone has opportunity if, at the time of the crime, AND if they can reasonably reach
% the place of murder.
opportunity(X) :- person(X), time_possibility(X), place_possibility(X).

% Someone also has opportunity if they were in the room with the victim at the time of death,
% and the murder method is poison. The idea is that the culprit could have theoretically administered
% the poison in the same room as a crowd without anyone noticing.
opportunity(X) :- cause_of_death(poison), time_of_crime(T), place_of_crime(P), person(X), alibi(X, P, T).

% Let us continue with the first definition, with the lack of an alibi.
% The 'closed world assumption' is entirely sufficient for our purposes.
time_possibility(X) :- person(X), time_of_crime(T), not(alibi(X, _, T)).
```

Slika 1. Primjer pravila u Prologu

Kroz razvoj igre, pokazalo se kako se baza pravila i činjenica, uz čuvanje pravila za određivanje počinitelja, može koristiti i za čuvanje trenutnog stanja igre (tko se nalazi u kojoj sobi, kako su sobe povezane, koje objekte igrač ima, koji objekti imaju otiske prstiju, koje je trenutno vrijeme, je li zločin počinjen, itd.) Uz to, baza također ima nekoliko jednostavnih pravila koja se koriste za određivanje kretnje drugih likova (može se smatrati kao jedan jako jednostavan oblik umjetne inteligencije).

Na temelju ovoga, samu bazu pravila dijelimo na tri dijela:

1. Detektiv / pravila za dedukciju zločinca
2. Pravila za ponašanje likova
3. Pravila za praćenje stanja u igri

U nastavku ukratko opisujemo svaku od ove tri cjeline.

2.1 Detektiv

Glavna zadaća detektiva je odrediti tko je počinitelj.

Počinitelj je netko tko ima priliku počiniti zločin, i netko protiv koga postoje konkretni dokazi.

Netko ima priliku počiniti zločin ako:

- nema alibi za vrijeme zločina
- je viđen direktno prije zločina na lokaciji od koje je mogao doći do mjesta zločina s obzirom na to koliko je prije zločina bio viđen
 - Primjerice, ako je netko bio viđen u dnevnoj sobi u 20:21, vrijeme i mjesto zločina je u spavaćoj sobi na 3. katu u 20:22, i poznato je da je potrebno barem 4 minute da se dođe iz dnevne sobe u tu spavaću sobu, zaključuje se da nije moguće da je ta osoba počinio zločin.
 - Analogno vrijedi i za viđenja direktno nakon zločina.

Naravno, postoje i dodatni slučajevi koje će detektiv smatrati sumnjivima.

- Najvažniji od njih je slučaj kada detektiv, gledajući prijašnje objašnjenje položaje direktno prije i poslije zločina, shvati da nešto nema smisla.
 - Primjerice, ako je netko viđen prije zločina u 20:22 u dnevnoj sobi, a nakon zločina u 20:24 u spavaćoj sobi, a poznato je da je potrebno barem 5 minuta da se dođe iz jedne sobe u drugu, detektiv će zaključiti da je osoba iskoristila nekakav trik da skрати to vrijeme, i da se ne može oslanjati na alibi te osobe.
 - U ovom konkretnom projektu, taj „trik“ su tajni prolazi koji spajaju sobe.

Inače, položaji i vremena se utvrđuju kroz pravilo da ako su u nekoj sobi dvoje ili više ljudi, oni si međusobno osiguravaju alibi za to vrijeme.

Dokaz protiv nekog smatraju se:

- otisci prstiju na oružju kojim se počinio zločin, ili
- posjedovanje nečega što je moglo biti oružje kojim se počinio zločin u trenutku kada policija dođe

2.2 Pravila za ponašanje likova

Trenutno, baza pravila podržava jedno od dvije mogućnosti likovima – mogu ili ostati u sobi u kojoj se nalaze, ili otići u jednu od drugih soba.

Lik se može kretati dok god je živ, i dok god ima dostupnih izlaza.

- Primjerice, može se dogoditi da ih netko zaključa u sobu.

Likovi mogu ostati u sobi u kojoj se nalaze dok god ne paničare. Trenutno, likovi ulaze u stanje panike ako čuju nešto glasno – u ovom slučaju, pucanj.

U prijašnjem dijelu spomenuti su tajni putevi među sobama. U početku, likovi nisu svjesni ovih puteva. Međutim, ako vide nekoga kako prolazi ili izlazi iz tih puteva, oni će ih postat svjesni, početi koristiti, i kasnije spomenuti njihovo postojanje policiji (što će detektiv onda uzeti u obzir pri razmatranju alibija.)

Kada osobe uđu u sobu gdje se nalazi truplo, automatski se poziva policija. Soba u kojoj je truplo nađeno se smatra mjestom zločina, i na temelju te lokacije gradi se detektivovo razmišljanje. Ovo pravilo je dodatno prvenstveno kako bi sama igra bila zanimljivija – naime, igrač ima mogućnost prenijeti truplo u drugu sobu i na temelju toga možda izgraditi neki bolji alibi.

Naravno, ako ijedna osoba vidi kako neka druga osoba radi nešto ilegalno ili jasno implicirajuće u zločin (ako vide sam trenutak zločina ili vide osobu koja nosi truplo ili ima krv na sebi), oni postaju svjedoci. (I, u kontekstu tekstualne avanture, automatski dovode do igračeve propasti.)

2.3 Pravila za praćenje stanja u igri

U samoj implementaciji, sve informacije o tome gdje se igrač nalazi, gdje se svi likovi nalaze, o objektima koje je igrač skupio, je li žrtva živa ili mrtva, tipovi objekata, mogu li se objekti uopće uzeti, o sobama, kako su sobe međusobno povezane, jesu li određena vrata zaključana, itd.

Ovdje su isto definirana pravila koja omogućavaju igraču da nosi truplo.

U konkretnoj implementaciji tekstualne avanture, činjenice će se izvlačiti upravo na temelju ovih pravila, umjesto pretraživanja konkretnih objekata u memoriji.

3. Tekstualna avantura

Tekstualna avantura, nazvana *Moths in the Darkness*, je osmišljena kao klasična igra u kojoj igrač zadaje naredbe pomoću naredbenog retka. Radnja se odvija u velikoj kući na noć zabave koju je priredila žrtva. U kući se također nalazi 5 gostiju, slučajno izabranih ih postojeće liste. Igračev cilj je počiniti ubojstvo prije nego što zabava završi, i na način koji garantira da ga policija neće uhvatiti.

Igrač se može:

- kretati (u kardinalnim smjerovima ili navođenjem konkretnog imena sobe)
- gledati oko sebe
- razgovarati s drugim osobama
- uzeti predmete ili ostavljati ih u sobama (te istim mehanizmom micati tijela, ako je osoba mrtva)
- zaključati i otključati vrata (ako nađe ključ)
- misliti (što je zapravo sustav pomoći, kojem se igraču ukazuje trenutni položaj žrtve radi lakšeg igranja)
- provjeriti trenutno vrijeme
- čekati 1 ili više minuta
- počiniti ubojstvo
- sam prijaviti tijelo (ako ne žele čekati da ih netko od gostiju nađe i misle da su uspješno isplanirali ubojstvo)

Tok tekstualne avanture može se opisati na sljedeći način:

- Igra počinje. Vrijeme je 20:00. Zabava završava u 22:30, i do tada igrač mora počiniti ubojstvo.
- Igrač se kreće po kući, skuplja predmete i razgovara i s likovima. Svaka kretnja, razgovor i uzimanje predmeta traje točno 1 minutu. To znači da svaki „potez“ traje jednu minutu. Korisničko sučelje prikazano je na slici 2.
- Jednom kada je igrač u istoj sobi kao žrtva, uz preduvjet da imaju oružje, mogu počiniti ubojstvo.
- Ako netko drugi svjedoči ubojstvu, igrač je uhvaćen u djelu i igra odmah završava (detektiv se uopće ne poziva).
- U protivnom, kada je truplo nađeno, dolazi policija, i detektiv se služi pravilima kako bi uspostavio krivicu. Prikazano na slici 3.
- Igrač pobjeđuje ako se ne nalazi u listi mogućih počinitelja ili ako počinu ubojstvo i nitko ne nađe tijelo do 22:30.

```

===== ENTRANCE HALL =====

You stare at your own shadow. It's sharp, in spite of how dim the overall room is. Put your hand in front of your face,
and there's a good chance you might lose sight of it. The shadow's the only thing that'll keep you grounded. As the
house creaks, you feel as if the few bits of light move as well, ever so slightly. Perhaps the ground intends to open
and swallow everything whole?

A solitary dusty coat rests on its little rack in the corner of the room. A faint clink makes you look up. A chandelier
dangles over your head. It's electric, yet most of its bulbs have died. That's not even mentioning the ones that had
likely cracked long ago.

To the north is the Main Hall. To the east is the Lounge.

> go north

===== MAIN HALL =====

Your feet practically sink into the carpet covering the entirety of the main hall's massive floor. So massive the room
is, in fact, that in its mass it forgot to have substance. It's so empty that you can hear the faint whistle of the
draft coming from somewhere. Let your thoughts wonder too far, and you'll swear you're seeing ghosts.

Duke is here.

To the south is the Entrance Hall. To the east is the Game Room. To the west is the Dining Room. To the north is the
Gallery.

> go west

```

Slika 2. Korisničko sučelje

```

The ones who found the body are:
['Duke']

The police are called...
The detective arrives at the scene...
The first thing he needs to establish is OPPORTUNITY. Who had a chance to kill the victim?
The coroner has accurately estimated that the time of the murder was: 20:05.
The scene of the crime was at the library.

Let us look at everyone's last known alibis before and after the crime.
After speaking with the witnesses, it would seem that the timetable would be something like...

LAST TIME SEEN BEFORE, OR AT THE MOMENT OF CRIME
Main Hall 20:05 ['Duke', 'Dinah'] [ COULD NOT HAVE MADE IT TO THE CRIME SCENE ]
Bedroom 20:05 ['Elliot', 'Benny'] [ COULD NOT HAVE MADE IT TO THE CRIME SCENE ]
Main Hall 20:04 ['Player'] [ COULD NOT HAVE MADE IT TO THE CRIME SCENE ]

FIRST TIME SEEN AFTER THE MURDER
Main Hall 20:06 ['Duke', 'Dinah'] [ COULD NOT HAVE BEEN HERE IF HAD COME FROM THE CRIME SCENE ]
Lounge 20:06 ['Benny', 'Elena'] [ COULD NOT HAVE BEEN HERE IF HAD COME FROM THE CRIME SCENE ]
Lounge 20:08 ['Elliot'] [ COULD NOT HAVE BEEN HERE IF HAD COME FROM THE CRIME SCENE ]
Dining Room 20:14 ['Player'] [ COULD HAVE MADE IT FROM THE CRIME SCENE AFTER THE FACT ]

Based on the time and the locations, we can conclude that the ones with the opportunity are:
['Robert']

Great! You've managed to use a trick to give yourself an alibi for the crime!
Of course, the victim is always on the list when it comes to opportunity. Even if it seems unlikely, given everything.

Next, let us consider the means. Do we possess any hard evidence against anyone? What about the MEANS?

```

Slika 3. Detektiv dolazi na scenu

4. Implementacija

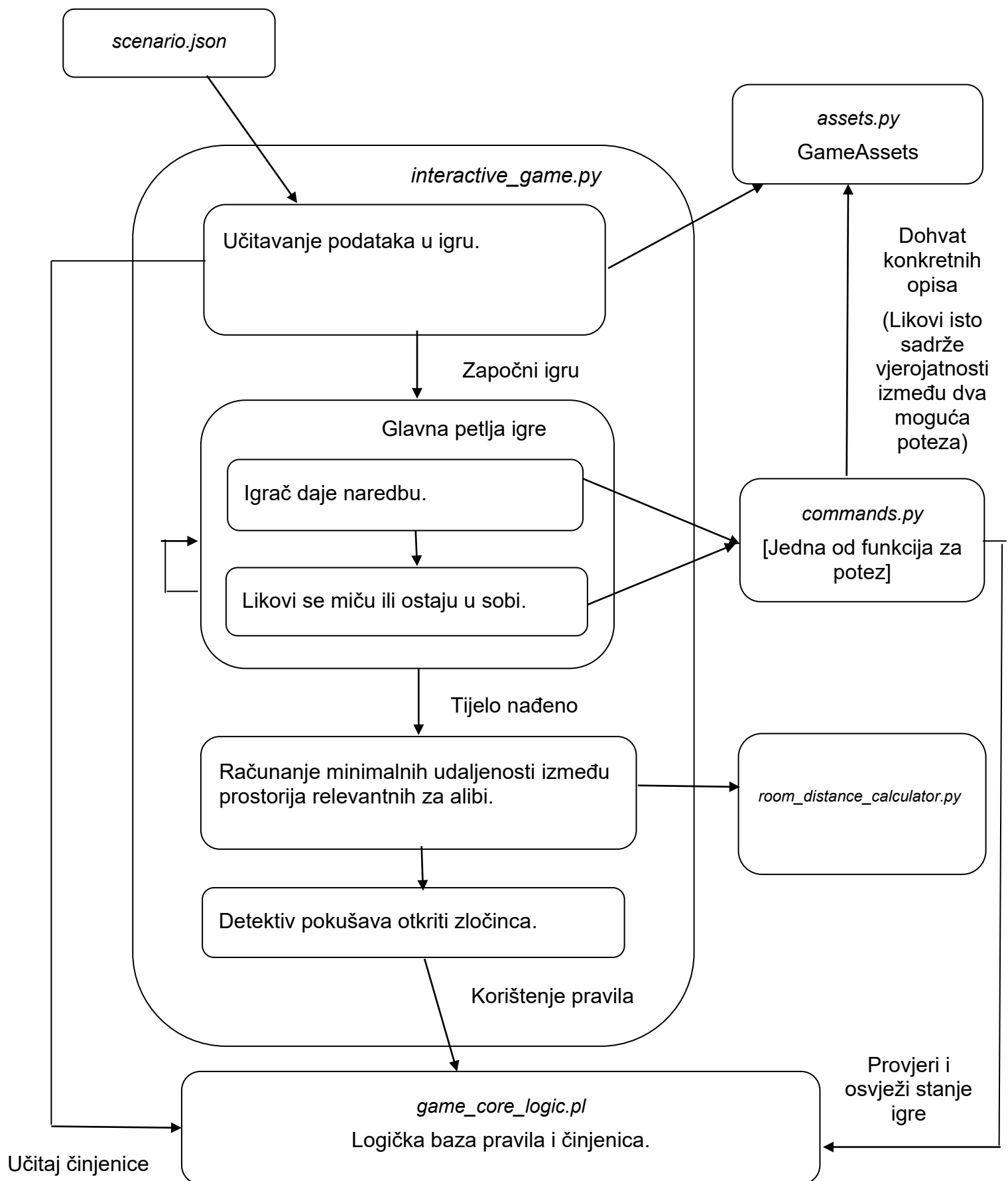
Tekstualna avantura implementirana je u Pythonu. Sučelje prema bazi pravila i činjenica u Prologu ostvareno je paketom *PySwip*. U pravilu, ovaj paket koristi se za SWI-Prolog; međutim, ako je složeni Prolog nije razmatrao u okviru ovog kolegija, držali smo se isključivo funkcionalnosti temeljnog Prologa.

Kod je ustrojen na način prikazan u Tablici 1.

Tablica 1. Struktura koda

scenario.json	Sadrži sve značajne informacije o igri – likovi, njihovi opisi, sobe, veze među njima, njihovi opisi, objekti – sve se to učitava u glavni program – opisi se učitavaju u Python strukture podataka, dok se osnovne informacije o odnosima prosljeđuju kao činjenice u Prolog bazu.
assets.py	Opisuje strukture podataka u koje se učitavaju konkretne informacije o sobama/objektima/likovima.
commands.py	Sadrži funkcije koje se pozivaju kada korisnik upisuje određene naredbe. Dio tih funkcija se koristi i za djelovanje likova koje korisnik ne kontrolira.
interactive_game.py	Sadrži samu igru – od inicijalizacije, do glavne petlje igre, do njenog završetka.
room_distance_calculator.py	U prijašnjem dijelu je objašnjeno kako se u zaključivanju koristi vremenska udaljenost među sobama (minute, odnosno potezi potrebni da se dođe od jedne sobe do druge.) Ta udaljenost se računa na način da se na temelju veza među sobama konstruira graf, te se zatim Bellman-Fordovim algoritmom taj graf minimizira, i određuje najkraći mogući put među sobama.
main.py	Glavni program.
game_core_logic.pl	Baza pravila.

Slika 4. vizualno prikazuje interakciju svih ovih elemenata.



Slika 4. Pojednostavljeni vizualni prikaz komponenti i funkcionalnosti

5. Instalacija i uporaba

Ovaj program zahtjeva:

- Python (3.6+)
- SWI-Prolog (32-bit ili 64-bit – ovisno o verziji Pythona koji se koristi!)
- Paketi dostupni u requirements.txt (pyswip, python-Levenshtein, termcolor, colorama)
 - Svi paketi mogu se instalirati odjednom koristeći naredbu: `pip install -r requirements.txt`

Igra se zatim pokreće naredbom:

```
python main.py
```

6. Opažanja, nedostaci i zaključak

Jedan od definitivnih izazova bio je u sastavljanju samih pravila. Bilo je pravila koja su se morala znatno pojednostaviti ili jednostavno izbaciti jer su ili previše usporavale sustav u trenutku evaluacije (primjerice, za pravila koja su koristila oznake vremena, pri evaluaciji bi se ispitale sve moguće oznake vremena – obično problem ako smo koristili atome koji nisu bili prisutni na „ONDA“ strani pravila), ili općenito prekomplikirana, do mjere gdje je bilo jednostavnije ručno postaviti nešto kao činjenicu (primjerice, umjesto pravila kojim bi se pratilo koji objekt ima nečije otiske prstiju, pokazalo se jednostavnije postaviti činjenicu da objekt ima otiske prstiju u samom kodu).

Često se problem sveo na kompromis – pravila su se obično mogla pojednostaviti ako smo pri svakom „potezu“ svakom stanju igre dodijelili vremensku oznaku; međutim, tu se pojavio problem što bi se strašan broj činjenica onda nakupio u bazi sa samo nekoliko poteza.

Primjerice, hoćemo li imati jedno pravilo koje će moći odrediti objekt Y kojeg osoba X ima bez obzira na vrijeme...

```
inventory(X, Y) :- ...
```

...ili ćemo za svaki trenutak igre imat zabilježeno da za taj trenutak vrijedi da je objekt Y u vlasništvu osobe X?

```
inventory(X, Y, T) :- ...
```

Drugo pravilo pojednostavljuje pravila i činjenice koje bismo morali koristiti, jer bi nas ionako pri upitima iz glavnog programa zanimalo stanje u trenutku T. Međutim, time znatno povećavamo broj činjenica!

Na kraju, nije bilo neko očito rješenje za dilemu – za drukčija pravila se primjenjivao drukčiji princip, ovisno o tome kakvi su upiti iz vanjskih modula bili očekivani.

Čak i uz taj trud, međutim, igra od vremena do vremena počne usporavati, obično nakon ubojstva. Točno pravilo koje usporava igru, nažalost, nije nađeno. Korištenje „cut“ simbola -- ! – bilo je isprobano, ali nije dalo nikakva poboljšanja; u nekim slučajevima je dovelo i do krivih rezultata, jer se primijenio u slučajevima gdje prestanak u traženju rezultata nije bio dopušten. Stoga u konačnici nije ni korišten.

Sve u svemu, međutim, sama igra ima potencijala, i sam ekspertni sustav radi kako je zamišljen.

7. Literatura

[1] SWI Prolog Reference Manual, https://www.swi-prolog.org/pldoc/doc_for?object=manual, 26.12.2020.

[2] Tekol Y. i doprinositelji, *PySwip*, <https://github.com/yuce/pyswip>, 2020.