

Entrópia Samu Programozói Kézikönyv

ESPORT ÉS MESTERSÉGES INTELLIGENCIA

Ed. ESAMU, Entrópia Samu,
2016. okt. 26, v.
prog.guide.hu.0.0.11.11

Copyright © 2016 Dr. Bátfai Norbert

Entrópia Samu

Copyright (C) 2016, Norbert Bátfai. Ph.D., batfai.norbert@inf.unideb.hu, nbatfai@gmail.com

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Ez a program szabad szoftver; terjeszthető illetve módosítható a Free Software Foundation által kiadott GNU General Public License dokumentumában leírtak; akár a licenc 3-as, akár (tetszőleges) későbbi változata szerint.

Ez a program abban a reményben kerül közreadásra, hogy hasznos lesz, de minden egyéb GARANCIA NÉLKÜL, az ELADHATÓSÁGRA vagy VALAMELY CÉLRA VALÓ ALKALMAZHATÓSÁGRA való származtatott garanciát is beleértve. További részleteket a GNU General Public License tartalmaz.

A felhasználónak a programmal együtt meg kell kapnia a GNU General Public License egy példányát; ha mégsem kapta meg, akkor tekintse meg a <http://www.gnu.org/licenses/> oldalon.

<http://gnu.hu/gplv3.html>

COLLABORATORS

| | | | |
|---------------|--|-------------------|------------------|
| | TITLE : Entrópia Samu Programozói Kézikönyv | | |
| <i>ACTION</i> | <i>NAME</i> | <i>DATE</i> | <i>SIGNATURE</i> |
| WRITTEN BY | Bátfai, Norbert, Besenczi, Renátó, és Simkó, Szilárd | 2016. november 5. | |

REVISION HISTORY

| NUMBER | DATE | DESCRIPTION | NAME |
|-------------|------------|---|-----------|
| 0.0.1 | 2016-08-23 | Iniciális dokumentum. A kiinduló 5let (eBudapest2024) sportvezetés eredményeket kérdező levelére esetleges válaszul elküldve, illetve a potenciális érdeklődőknek/befektetőknek elküldve. | nbatfai |
| 0.0.2 | 2016-08-25 | Javítások. | nbatfai |
| 0.0.3 | 2016-09-02 | Továbbfejlesztés: a sztenderd és a váratlan megoldási irány felvázolása. A DevRob2Psy-ba megosztva. | nbatfai |
| 0.0.4 | 2016-09-03 | Javítások. A DevRob2Psy, UDPROG-ba megosztva. | nbatfai |
| 0.0.5 | 2016-09-04 | Javítások. A játékelmény. A DevRob2Psy, UDPROG-ba megosztva, e-mailben szétküldve potenciális érdeklődőknek. | nbatfai |
| 0.0.6 | 2016-09-05 | Technikai okokból LibreOffice-ból Word-be (ha lesz érdeklődés, minden további dokumentáció már DocBook XML-ben). A doksi újra szétküldve. | nbatfai |
| 0.0.7 | 2016-09-14 | DocBook XML 5.1 átirat és a kutatási terv elkezd. | nbatfai |
| 0.0.8 | 2016-09-15 | Iniciális doksi a GitHub-ra, innentől a verziókezelés majd az ottani git-el, lásd majd a changelog-ot: https://github.com/nbatfai/SamuEntropy | nbatfai |
| hu.0.0.10.8 | 2016-09-26 | Google Firebase rendszer licencmérnökségi feladat kifejtése. | rbesenczi |

| REVISION HISTORY | | | |
|------------------|------------|--|-------------|
| NUMBER | DATE | DESCRIPTION | NAME |
| hu.0.0.11.7 | 2016-10-19 | Google Firebase-t használó gyors prototípus linkelése. | viktorsimko |

Tartalomjegyzék

| | |
|--|-----------|
| I. Szójegyzék | 1 |
| II. Bevezetés | 4 |
| 1. Bevezetés | 6 |
| 1.1. Az eredeti 5let | 6 |
| 1.1.1. Az 5let felvetése | 6 |
| 1.1.1.1. A projekt célok | 6 |
| 1.1.1.2. A projekt rész céljai | 6 |
| 1.1.1.2.1. Esport | 6 |
| 1.1.1.2.2. Történelemtudomány | 6 |
| 1.1.1.2.3. Neveléstudomány, pszichológia | 7 |
| 1.1.1.2.4. Esport élettan | 7 |
| 1.1.1.2.5. Promóció és közösségépítés | 7 |
| 1.1.1.2.6. Mesterséges intelligencia | 7 |
| 1.1.2. Egy várható megvalósítás | 7 |
| 1.2. Esport és mesterséges intelligencia | 7 |
| 1.2.1. Egy váratlan megvalósítás | 7 |
| 1.2.1.1. Miért szeretünk játszani? | 7 |
| 1.2.1.2. Családi robotika és játék | 8 |
| 1.2.1.3. A recept | 8 |
| 1.2.1.4. A játékelmény | 9 |
| 1.2.1.5. Kockázatok | 9 |
| 1.2.2. Összefoglalás | 9 |
| III. Kutatási terv | 10 |
| 2. Bevezetés | 12 |
| 2.1. Bevezetés | 12 |

| | |
|--|----|
| 2.2. Entrópia Samu | 12 |
| 2.2.1. A család | 12 |
| 2.2.2. Főbb használati esetek | 13 |
| 2.2.2.1. Samu, az agy | 13 |
| 2.2.2.2. Gréta, az építő | 13 |
| 2.2.2.3. Nándi, a tanító | 13 |
| 2.2.2.4. Matyi, a vadász | 13 |
| 2.2.2.5. Erika, a harcos | 13 |
| 2.2.2.6. Norbi, a törzsfőnök | 13 |
| 2.3. Oktatási és kutatási terv | 13 |
| 2.3.1. Gyors prototípusozás | 14 |
| 2.3.1.1. Samu Family Album. | 14 |
| 2.3.1.1.1. Samu | 14 |
| 2.3.1.1.2. Gréta | 14 |
| 2.3.1.1.3. Nándi | 14 |
| 2.3.1.1.4. Matyi | 14 |
| 2.3.1.1.5. Erika | 15 |
| 2.3.1.1.6. Konkrét app 5letek | 15 |
| 2.3.1.1.6.1. Norbi | 15 |
| 2.3.1.1.6.2. Samu | 15 |
| 2.3.1.1.6.3. Gréta | 16 |
| 2.3.1.1.6.4. Az 5. védés | 17 |
| 2.3.1.1.6.5. Nándi | 17 |
| 2.3.1.1.6.6. Matyi | 17 |
| 2.3.1.1.6.7. Erika | 18 |
| 2.3.1.1.6.8. Játékelmény és vizualizációs kísérletek | 18 |
| 2.3.1.2. Samu OOCWC Fight | 18 |
| 2.3.1.3. Samu DevRob Vision | 18 |
| 2.3.1.4. Samu Custom RTS | 18 |
| 2.3.2. Szakfordítói jellegű feladatok | 18 |
| 2.3.3. Licencmérnökség jellegű feladatok | 18 |
| 2.3.4. Kreatív és művészeti jellegű feladatok | 19 |
| 2.3.4.1. Logók | 19 |
| 2.3.4.2. Konkrét app 5letek | 19 |
| 2.3.5. Promóciós jellegű feladatok | 19 |
| 2.4. Köszönetnyilvánítás | 19 |

Ábrák jegyzéke

| | |
|---|---|
| 1.1. Pillanatkép a 0 A .D. játék han_china modjában a szerző (a térképen piros) által játszott egy játékmenetből. . . . | 8 |
|---|---|

I. rész

Szójegyzék

D

Debreceni Egyetem [DE]

A DE a Debreceni Egyetem rövidítése.

Developmental Robotics 2 Robopsychology [DevRob2Psy]

A DevRob2Psy a Developmental Robotics 2 Robopsychology nevű Facebook csoport rövidítése. Ez egy informális csoport, amely a címbeli témák iránt érdeklődőket tömöríti, tagjai főleg a DE hallgatói és oktatói. Ebben a csoportban kommentelheti a kedves olvasó ezt a doksit is: <https://www.facebook.com/groups/devrob2psy/>.

lásd még "**UDPROG**".

E

Entrópia Samu [ESAMU]

Az esport és a mesterséges intelligencia között nyilvánvaló a szoros kapcsolat, melyet a jelen kutatásunkban még tovább akarunk fokozni azzal, hogy nem csupán a játékokbeli MI felhasználásra gondolunk, hanem magának az MI-nek a megteremtését várjuk az új, fejlesztendő esporttól. Az aktuális elképzelés szerint ez egy szoftveresen implementált, a családi PC-n futó, mobil eszköz érzékszervekkel ellátott fejlődésrobotikai és robotpszichológiai Samu alkalmazás lesz: az Entrópia Samu, <https://github.com/nbatfai/SamuEntropy>.

lásd még "**Samu**".

F

Fejlődésrobotika [DevRob]

Olyan tudományos irányvonal, amely a testtel rendelkező robotok tanulását az emberi tanulás szemszögéből vizsgálja. Alaptézise, hogy a robot testtel rendelkezik, ennek antézise lehet a csak szoftveres robot, a kettő szintézisét a PSAMU beküldött kéziratban adtuk meg.

lásd még "**Robotpszichológia**".

I

IEEE Recommended Practice for Software Requirements Specifications (SRS) [IEEE Std 830-1998]

A szoftverkövetelmény dokumentum [IEEE szabványa](#). Entrópia Samu szoftverkövetelmény doksját ennek a szabványnak megfelelően készítjük majd el.

J

JIBO

Cynthia Breazeal társasági robotja, lásd <https://www.jibo.com/>

R

Robotpszichológia

Az [Asimovi robotpszichológia](#) programozói szemszögből értelmezett implementálása, lásd <https://github.com/nbatfai/-Robopsychology>.

S

Samu

Egyszer a fejlődésrobotikus családi csevegőrobottal, Bátfai Samuval, másrészt általában a mesterséges intelligencia fejlődésrobotikai megközelítésével kapcsolatos kutatásaink összefoglaló neve. Lásd még a <https://prezi.com/utlu1bevq9j2/egy-testetlen-fejlodesrobotikai-agens/> prezit és a <https://arxiv.org/abs/1511.02889> illetve a PSAMU beküldött kéziratot.
lásd még "[ESAMU](#)".

U

The Yearbook of the Programmers of University of Debrecen [UDPROG]

A Debreceni Egyetem [reguláris programozás oktatásához](#) kapcsolódó, LinkedIn-en szervezett [fejlesztői közösség](#), melynek egyik gyűjtőpontjában [szoftverek](#), a másikban maga az [évkönyv](#) áll.

II. rész

Bevezetés

Először bemutatjuk az eredeti 5letet, hangsúlyozva annak interdiszciplináris (például esport, mesterséges intelligencia, neveléstudomány, pszichológia, történelem, és informatika) jellegét. Majd az 5letet két irányban bontjuk ki, egy várhatóban, amelyre mindenki természetesen gondolna, illetve egy kevésbé magától értetődőben, ahol már nem (csak) a játék a fejlesztendő esport célja, hanem a mesterséges intelligencia kutatás sodorvonalába való bekapcsolódás, ahol az esport már nem csupán használja az MI-t, hanem megpróbálja megteremteni annak egy új minőségét.

1. fejezet

Bevezetés

1.1. Az eredeti 5let

Vázoljuk a kiinduló 5letet és annak egy magától értetődő megvalósítási irányát. A tárgyalásban lényegesen kitágítjuk az eredeti 5let koncepcióját és már nemcsak esport játékként fogalmazzuk meg, hanem egy nem triviálisan adódó megvalósítási irányként, mint alapkutatási célt is vázoljuk.

A jelen dokumentum célja a **kutatási terv** elkészítésének megalapozása.

1.1.1. Az 5let felvetése

Az 5let munkacíme ez volt: „Aki mer, az nyer” e-Budapest 2024, alcíme: *esport kutatási projekt javaslat, avagy „az esport napjaink sakk-tömegsportja”?*

1.1.1.1. A projekt célok

A tervezett projekt célja egy „saját ” interdiszciplináris (elsődlegesen esport, mesterséges intelligencia, neveléstudomány, pszichológia, történelem, és informatika) esport torna kialakításának és a professzionális esportolók (elsődlegesen esport élettani¹ és informatikai²) támogatási lehetőségeinek kutatása.

1.1.1.2. A projekt rész céljai

1.1.1.2.1. Esport

A projekt fő célja több nyílt forráskódú számítógépes (tipikusan RTS³) játék megvizsgálása, hogy alkalmasak lennének-e egy ráépülő “mozgalom” és esport viadal kialakításához és fenntartásához. A további rész célokat már egy konkrét ilyen példa mentén vesszük sorra, ennek megfelelően előzetesen essen a választásunk a 0 A. D.⁴ nyílt forráskódú, korhű történelmi RTS játékra.

1.1.1.2.2. Történelemtudomány

Konkrét rész cél lehet saját, korhű modok⁵ kialakítása, például a honfoglaló magyarságé.

¹ Ez egy érintetlen területnek tűnik, amely viszont akár beilleszthető lehet a DE egyébként stratégiai egészségipari törekvéseibe.

² Például adott játék esetén stratégiák, forgatókönyvek feltárása, elemzése.

³ Real Time Strategy, valós idejű stratégiai játék

⁴ A 0 A. D. egy nyílt forráskódú, valós idejű stratégiai játék, <https://play0ad.com/>, <https://github.com/0ad/0ad/>.

⁵ Egy aktuális mod például a szóban forgó játékhoz a Han-dinasztiát megvalósító: <http://www.moddb.com/mods/rote>, https://github.com/0ADMods/-han_china.

1.1.1.2.3. Neveléstudomány, pszichológia

Az oktatási rendszer kontextusában a játék (tipikusan gyerekekre alapuló) tömegbázisa természetes módon vetne fel neveléstudományi, pszichológiai kérdéseket.

1.1.1.2.4. Esport élettan

Az előző pont említette tömegbázison vizsgált kérdések önmagukban fontosak, de vizsgálatuk kamatoztatható lenne a professzionális játékosok támogatására is? A professzionális játékosok extrémebb igénybevételének kutatása, a játékosok (például klasszikus élettani) támogatása direkt természetes cél lehet. Ugyanúgy érdekes a tömegsport (és annak vélt vagy valós „állandóan a képernyőt bámulod” jellegű túlkapásainak) vizsgálata.

1.1.1.2.5. Promóció és közösségépítés

Adott esetben a bevezetendő torna kapcsán legígéretesebb játék lehetne a Budapest 2024 egy „nem hivatalos”⁶ játéka is.

1.1.1.2.6. Mesterséges intelligencia

Az aktuális Google DeepMind Ataris⁷, a Nature folyóiratban megjelent cikke egy forró témát nyitott a mesterséges intelligencia kutatásokban, amire figyelni, amihez kapcsolódni teljesen természetes és fontos.

1.1.2. Egy várható megvalósítás

A bevezetésben használt 0 A. D. játékra vagy hasonlóra épített vertikális kutatási irány egy várható (sztenderd) megvalósítási irány, amelyben jó eséllyel valóra tudnánk váltani az említett esport, mesterséges intelligencia, neveléstudomány, pszichológia, történelem és informatikai célokat, de – vagy éppen ezért – a következő tárgyalásban egy másik, egy kevésbé előre megjósolható ám még ambiciózusabb megoldási irányt vázolunk fel.

1.2. Esport és mesterséges intelligencia

1.2.1. Egy váratlan megvalósítás

A játékfejlesztés (és általában a szoftverfejlesztés) Szent Grálja a „Killer App”⁸-ok keresése. Ezeknek az app-oknak a ritkaság az alaptermészete, killer app-ot csinálni várható megvalósítással lehetetlen. Persze közel ugyanez egy nem várható megvalósítási irányra is igaz lenne, mert a killer app-okat nem csinálják, hanem egy adott app válhat azzá, HA.

Ez esetben viszont – legalábbis a kutatásban – bátran megpróbálhatjuk „megváltani a világot”. Ehhez először nézzük meg, hogy miért szeretünk játszani, majd azt, hogy egy fejlődésrobotikai esport játékkal ki tudnánk-e fejleszteni mesterséges intelligenciát? Tehát itt már nem a gépi játékos vezérlését vagy a játékmenet vezérlését értjük mesterséges intelligencia alatt, hanem egy intelligens új életforma megteremtését, ez persze most egy teljesen intuitív cél, tehát a teremtetendő létforma jelentsen most még bármit.

1.2.1.1. Miért szeretünk játszani?

Fizikai testünk a kémia „rendezetlen” molekuláris világában él, de mint élő rendszer képes növelni és fenntartani belső rendezettségét (azaz felnőni és élete során őrizni átmeneti állandóságát) ebben a világban⁹. Feltételezhetjük, hogy ezt a tulajdonságot

⁶ A nem hivatalos titulus azt szeretné érzékeltetni, hogy a bevezetendő játék még nem lenne olimpiai sport :), de nyilván egy lehetséges lépés lenne ebbe az irányba. Azt mindenesetre megjegyezhetjük, hogy az esportnak olimpiai sportként való elfogadtatásával – Rió kapcsán például – napi szinten olvashatunk híreket (lásd például a “League of Legends” játékot).

⁷ <https://storage.googleapis.com/deepmind-data/assets/papers/DeepMindNature14236Paper.pdf>

⁸ A „killer app” fogalom érzékeltetésére például a Doom az volt a PC-s játékok világában vagy jóideje a Facebook az a közösségi portáloknál.

⁹ Lásd Erwin Schrödinger, What is life? : the physical aspect of the living cell, Cambridge University Press, 1944 illetve Roger Penrose, A császár új elméje Számítógépek, gondolkodás és a fizika törvényei, Akadémiai, Budapest, 1993.

a mentális működéseink is „öröklük” abban az értelemben, hogy az érzékszerveink felfogta zajos inputba képesek olyan rendet beleláttni, amelyet mi úgy nevezünk: a valóság. *Tézisünk, hogy azért szeretünk játszani, mert a játékban a játék világának belső szervezettségét növelni tudjuk.*

Ezt a tézist szubjektív élményeimmel és a gyermekeimmel történt beszélgetések alapján tudom csupán megerősíteni, vagy még inkább csak illusztrálni. Például a 0 A. D. adta üres világot mindinkább megszervezni (beépíteni) személyes tapasztalatom szerint ez egy nagyon jó játékelményt adó elfoglaltság. Ennek egy pillanatfelvételét látjuk a következő ábrán.



1.1. ábra. Pillanatkép a 0 A. D. játék han_china módjában a szerző (a térképen piros) által játszott egy játékmenetből.

Játékelményt alatt azt értjük, hogy a játék fejlesztői alapvetően egy élményt kódolnak a játékkal, amelyet a játékosok dekódolni tudnak, azaz át tudnak élni¹⁰.

Sőt gyermekeimnél volt olyan életkor, amikor csatázni nem is, csak építeni (a növekvő Town Hall szint adta új építményeket elhelyezni és a játékbeli szereplőket „kimaxolni”, akartak. Ebben az időben éppen ezt az élményt adta a Minecraft® játék¹¹ survival módjában¹² a szabad építkezés.

További látványos példaként említjük a szervezettség növelésére a Clash of Clans® című játékot¹³, ahol a játékelményt hasonló. S ha a kedves olvasó játssza is esetleg ezt a játékot, akkor maga is megnézheti (rákeresve például a „Bátfai clan” nevű klánra) hogy a „th”-k¹⁴ változása jól mutatja a játék világa falvainak növekvő komplexitását, a szervezettség növekedése a klántagok faluin (a klánbeli helyezések sorrendjében, ami durván a játékra fordított idővel korrelál) kézzel fogható.

1.2.1.2. Családi robotika és játék

Szoftveresen implementált, a családi PC-n futó, mobil eszközös érzékszervekkel felszerelt, JIBO-jellegű fejlődésrobotikai (Samu) alkalmazás fejlesztését már korábban felvetettük és a DevRob2Spy kutatási terveinek része¹⁵. De hiányzott még a motiváció, hogy miért is használná ezt bárki is a családban? Erre lehet egy válasz a játék!

1.2.1.3. A recept

- Adjuk meg az önmaga belső szervezettségét növelni tudó gép modelljét¹⁶.

¹⁰ Lásd ennek kapcsán a következő publikációt: Bátfai Norbert, Bátfai Erika, A mobil játéfejlesztés elméleti és gyakorlati momentumai, Híradástechnika, 5, 34-37, 2005, http://www.hiradastechnika.hu/data/upload/file/2005/2005_5/HT_0505-7.pdf.

¹¹ <https://minecraft.net/>

¹² A játék egyfajta „örök élet”, módja, ahol szabadon lehet építeni, nem kell tartani semmitől.

¹³ <http://supercell.com/en/games/clashofclans/>

¹⁴ Ez a Town Hall szintjének rövidítése.

¹⁵ Lásd a Samu as a social robot for the family's home desktop (PC/Samu mint társas robot a család asztali gépén) kutatási témát és az N. Bátfai, R. Besenczi, Robopsychology Manifesto: Samu in His Prenatal Development (submitted manuscript) kéziratot.

¹⁶ Ez sajnos egy előre nem tervezhető lépés (Gábor Dénes után szabadon: ezt előbb még fel kell találni :) lásd a Norbert Bátfai, The Fractal Nature of Algorithms, <https://github.com/nbatafai/AlgorithmicFractals> c. kéziratot.

- Ezt a modellt valósítsuk meg egy “szoftveresen implementált, a családi PC-n futó, mobil eszközös érzékszervekkel felszerelt, JIBO-jellegű fejlődésrobotikai alkalmazás” formájában.
- Az alkalmazás belső szerkezetének egy vizuális interfésze lehetne egy játék alapja.

1.2.1.4. A játékélmény

Ebben a pontban kísérletet teszünk a játékélmény megálmodására. Két alapvető részből gyúrnánk össze:

- legyen a játék világának szervezettséget növelő része
- és legyen közösségi része.

Miközben a játék célja legyen az, hogy Samu (avagy a „fejlődésrobotikus tanuló gép”) minél jobban lásson és halljon!¹⁷

A szervezettséget növelő részt a nehezebb kitalálni, mert ez függ az előző pont 1-es pontjától („Adjuk meg az önmaga belső szervezettségét növelni tudó gép modelljét.”), amely önmagában egy nagy tudományos kihívás... tehát most maximum intuitív analógiákat vonhatunk:

- Ha a szóban forgó „entrópia-csökkentő gép” fogalma a Turing gép fogalmához hasonlatos lenne, akkor a GoDNGoD c. játéktervhez analóg módon tudnánk vizualizálni a definíciót. (Az említett játékban a Turing gép fogalmon alapuló gép fajtákat – breed-eket – szaporítunk, mint például a “Breedi Turingi”-t vagy a “Turingus Tri Breedus”-t, lásd a Norbert Bátfai, Turing’s Imitation Game has been Improved, <http://arxiv.org/abs/1509.00584> c. kéziratot).
- Ha a játékban a Samu látó (halló) megvalósítása használna több rejtett réteges neurális architektúrát, akkor arra egy vizuális nyelv¹⁸ megadható, amellyel a játékosok hangolhatják az architektúrát. Ha például ez a neurális architektúra megvalósítás a Google DeepMind TensorFlow¹⁹-ján alapulna (ami egy „vizuális nyelvből” képes C++ forrást generálni például) akkor annak folyamatára nyelvére egy „játékosabb front-end” nyelv kifejlesztése lehetne egy kutatási rész cél. Ha ez járható út lenne, akkor maga a játék lehetne a játékos saját digitális élőlényei „agyának” olyan elrendezése, amelyek minél jobb látást adna a lénynek. Az erre épülő közösségi szinten pedig vagy „harcolhatnak” ezek a lények, hogy melyik lát jobban, vagy együttműködhetnek, miszerint két játékos „rekombinálná” a lényeit, ahol kérdés, hogy az eredmény lény hogyan lát.
- A „brain storming”-ot bár folytathatnánk, de itt nem tesszük.

1.2.1.5. Kockázatok

Egyértelmű, hogy a „váratlan” irány alapkutatás és számos olyan buktatót tartalmaz, amelyek már alkalmazott kutatási célok kitűzését és megvalósítását is meggátolhatják, nem is beszélve az alkalmazott kutatásra esetleg építhető kísérleti fejlesztésről.

1.2.2. Összefoglalás

Kiindulásként az aktuális Budapest 2024 sportdiplomáciai erőfeszítésekre is hajazva vázoltunk egy esport kutatási tervet, amelyet alap- és alkalmazott kutatási irányokba álmotunk tovább. Miközben az alapkutatási irányban egy hard-core mesterséges intelligencia kutatási fókuszot is helyeztünk.

Ha sikerül forrásokat találni a kutatáshoz, akkor mindkét, a „várható” és a „váratlan” irányokon is érdemes dolgozni párhuzamosan, hiszen az előbbi alkalmazott kutatás lévén kézzelfogható eredménnyel kecsegtet.

Támogatás nélkül csak az alapkutatás javasolt.

Mindenesetre a futó Samuval kapcsolatos fejlődésrobotikai és robotpszichológiai kutatásainkat a jelen dokumentumban vázolt esport koncepcióba ágyazzuk majd be a továbbiakban.

A következőkben a a „várható” kutatási irányt a *Szabvány Samu*, a „váratlan” kutatási irányt az *Entrópia Samu* munkanévvvel illetjük.

¹⁷ Csak intuitív értelemben említhetjük ide a kapcsolódó kísérleteinket. A “látásra” például: <https://github.com/nbatfai/SamuCam>, a “hallásra”: <https://github.com/nbatfai/SamuVocab>.

¹⁸ Gondolatébresztőnek lásd például a korábbi LEGO Mindstorms®-hoz az MIT média laborja által fejlesztett nyelvet, Fred G. Martin, The Art of LEGO Design, The Robotics Practitioner: The Journal for Robot Builders, 1(2), 1995. <http://www.kipr.org/sites/default/files/artoflego.pdf>.

¹⁹ <https://www.tensorflow.org/>

III. rész

Kutatási terv

Ebben a részben kísérletet teszünk az *Entrópia Samu* kutatási tervének összeállítására.

Az esport és a mesterséges intelligencia között nyilvánvaló a szoros kapcsolat, melyet ebben a kutatásunkban még tovább próbálunk fokozni azzal, hogy nem csupán a játékokbeli MI felhasználásra gondolunk, hanem magának az MI-nek a megteremtését várjuk az új, fejlesztendő esporttól. Az aktuális elképzelés szerint ez egy szoftveresen implementált, a családi PC-n futó, mobil eszköz érzékszervekkel ellátott fejlődésrobotikai és robotpszichológiai Samu alkalmazás lesz: az Entrópia Samu, lásd: <https://github.com/nbatfai/SamuEntropy>.

Ennek a résznek a feladata a fejlesztendő esport játék kiválasztása, majd annak SRS doksija készítésének támogatása.

2. fejezet

Bevezetés

2.1. Bevezetés

Több alkalommal tűnt már úgy, hogy elérhető a mesterséges intelligencia létrehozása, de a siker csak nem jött el. A jelenlegi várakozásokat a Google DeepMind eredményei hízlalták fel. Samu is részben ennek inspirálására és ebbe az irányba tett törekvés, de lehet hasztalan. Ezért most nem is a Samus irányainkra koncentrálunk, hanem egy egészen más perspektívából közelítünk a kérdéshez: megálmodunk egy olyan esport játékot, amelynek célja mesterségesen intelligens faj létrehozása, ahol a faj egyelőre legyen csak egy intuitív megjelölés, jelentsen a későbbiekben bármit.

2.2. Entrópia Samu

Ebben a pontban tehát meg kell álmodunk egy játékot! Ezt egy eldobható gyors prototípus formájában tesszük meg, hogy szerezzünk némi tapasztalatot.

Hol játszódik a játék? A Földön. Mikor? Napjainkban.

2.2.1. A család

A játék szervezésének alapja a család. Család alatt azt a közösséget értjük, amely gondoz egy Samu ágenszt. Család (pontosabban a fejlesztendő játékbeli családot fenntartó) lehet egy a szó soros értelmében vett család, egy baráti társaság vagy egy egyetemi kurzus közössége, de akár egyetlen humán játékos is.

Architektúrális értelemben a családot az agynak (az ágens agyának) tekinthetjük (ami például egy PC-n vagy laptopon futó program) amelyhez érzékelő és mozgató részek csatlakozhatnak (melyekre mint mobilos érzékszervekre vagy izmokra gondolhatunk).

Funkcionális értelemben az agy végzi a számításigényesebb tevékenységeket, még a tipikusan mobil eszközön futó programok az érzékelést és a mozgatót végzik. Konkrétan egy humán játékos mobil kamerája lehet az agy szeme, GPS helymeghatározása egy újabb érzékszerve. Samu mozgása pedig azon keresztül valósul meg, hogy a mobil tulajdonosa viszi a mobilt. Gondoljunk arra, hogy ha az agy kíváncs, mi történik valahol, akkor a játékos a mobilja GPS-ének segítségével odamegy és kamerájával fotót készít a helyszínről, lásd majd a Matyi, a vadász használati esetet.



A család az architektúrális keret

Intuitívan gondolj arra, hogy a fejlesztendő játékbeli család fogalmunk az imént vázolt architektúrát rögzíti.

2.2.2. Főbb használati esetek

A következő használati esetek nem egy konkrét játékot specifikálnak, hanem csak adott irányokat jelölnek ki (amelyeket a fejlesztendő konkrét játékoknak meg kell próbálniuk megvalósítani), majd a **Gyors prototípusozás** c. pontban vezetünk be konkrét játékokat. Ezek a használati esetek a fejlesztendő alkalmazások archetípusait jelölik ki.

2.2.2.1. Samu, az agy

Ebbe a csokorba olyan programok tartoznak majd, amelyek gépi tanulós algoritmusok megvalósításai.

2.2.2.2. Gréta, az építő

Ebbe a csokorba olyan programok tartoznak majd, amelyek vizuális interfészt adnak a Samu az ágens pont programjaihoz. Fontos, hogy ezeknek a programoknak a rendezettség növelése játékelményt kell adniuk.

2.2.2.3. Nándi, a tanító

Ebbe a csokorba olyan programok tartoznak majd, amelyek felügyelt tanulást biztosítanak.

2.2.2.4. Matyi, a vadász

Ebbe a csokorba olyan programok tartoznak majd, amelyek az érzékelést és a mozgást valósítják meg (adott esetben például geocaching jelleggel).

2.2.2.5. Erika, a harcos

Ebbe a csokorba olyan programok tartoznak majd, amelyek aspirálnak esportként is üzemelni.



Milyen a jó Erika, a harcos program?

Intuitívan gondoldj arra, hogy úgy lehessen kivetítve játszani egy nagy arénában, mint egy nagy box gála fő mérkőzését. Tehát nem egy tipikus kvíz játékokra gondolunk itt... Akkor lehet jó a játékelményt, ha nézni is jó!

2.2.2.6. Norbi, a törzsfőnök

Ebbe a csokorba olyan szerveroldali programok tartoznak majd, amelyek a családok és az agycsaták kezelését valósítják meg.

2.3. Oktatási és kutatási terv

Ezekon a feladatokon egyaránt dolgoznak oktatók és hallgatók. Az aktuális hallgatóknak érdekes, hogy ezek az UDPROG (Dev-Rob2Psy, UDRFT) közösségben is kidolgozható feladatok, ilyen értelemben tekinthetjük ezeket felhasználhatónak az oktatásban.

A feladatok megoldásának menete a következő:

- i. Forkold le ezt a <https://github.com/nbatfai/SamuEntropy> projektet!
 - ii. A forkodban készítsd el a megoldásodat!
 - iii. Küldj egy pull request-et! (az aktív UDPROG hallgatóknak a feladat pontértékének fele plusszban elszámolható, ha a pull request merge-ölésre kerül, ez nyilván egy szubjektív döntés eredménye, ezért tekintsük egyfajta bónusznak :)
-

2.3.1. Gyors prototípusozás

5letek és megvalósítások kerülnek ide. Fontos, hogy megfeleljenek a **Főbb használati esetek** c. pont Gréta, Nándi, Matyi, Erika bontásának. Ez egy brain storming jellegű pont, de annyiban több, hogy olyan konkrét játék 5leteket kell vázolni, amelyekre egy gyors protót meg is tudunk valóban gyors protóként valósítani és ezen megvizsgálni a **játékélményt**.

A konkrét játékoknál az agyviharban felmerülő fícsöröket érdekes külön is kiemelni, hátha azok egy másik játékba is bevezethetők.

**Meta-fícsör: egy agy mind felett**

Az egy külön érdekes irány lehet, hogy nem a legjobbnak vélt prototípust választjuk majd ki innen megvalósításra (SRS doksi elkészítése stb.) hanem ha jól funkcionál a **Főbb használati esetek** c. pont alatti Gréta, Nándi, Matyi, Erika archetikus bontásunk, akkor az agy eseti cseréjével a játékosok váltogathatnák, hogy éppen melyik játékban vesznek részt.

2.3.1.1. Samu Family Album¹.

Ennek a konkrét játéknak a lényege az arcfelismerés².

Minden családhoz tartozó humán játékosok fotókat (arckép) készítenek magukól, más családok tagjairól, esetleg a játékban nem résztvevőkről is.

2.3.1.1.1. Samu

A fejlesztendő ágens feladata arcok (a saját és más családok arcainak) felismerése. Egy megvalósításban használjuk a Google TensorFlow csomagját. Ez tipikusan PC-s app lenne ma.

2.3.1.1.2. Gréta

A fejlesztendő TensorFlow-os Samu megvalósítás feladata, a TensorFlow adatfolyam gráfhoz játékélményt adni képes vizuális interfész biztosítása (Android és PC-s app-ok is).

2.3.1.1.3. Nándi

A fejlesztendő program feladata az arcokhoz kapcsolt adatok (például kié az arc, milyen érzést fejez ki stb.) felügyelt tanítása.

Ha a „Matyi” funkcióban a játékos fotót is készít a vadászat során, akkor a humán „Nándi tanító” tanítja meg arra az agyat, hogy valódi-e a lefotózott játékos profilja.

2.3.1.1.4. Matyi

A fejlesztendő program feladata az arcképek és a kapcsolható adatok felvétele. Például a szerver jelez, hogy a közelben (OpenStreetMap és vagy Google Maps megjelenítés) egy regisztrált játékos, vagy Bluetooth-on keresztül elérhető (vagy más, szoftverek kívüli szervező jelet vesz észre a játékos, mondjuk bar- vagy QR kódos karkötő, kitűző esetleg NFC matrica stb.) akkor arcot és kísérő adatokat cserélnék.

**Fícsör: bitek gyűjtése**

A játékos a találkozásokra az sarkallja, hogy ezekért biteket kap! Általában is próbáljuk a játékon belüli fizetőeszközt, összemérési alapot, jószágot stb. információban, azaz bitekben megfogalmazni.

Például a vadászat a találkozásokról relatív gyakoriságot vezet (X-el x_1 -szer, Y-nal y_1 -szer, stb. addig ismeretlennel pedig ennyiszor találkoztam) ez alapján a következő találkozáskor a játékosok annyi bitet kapnak, amennyi a „kivel találkoztam” entrópiája (amit esetleg csak akkor kap meg, ha prediktált és eltalálta stb).

¹ Munkacímnek: ARC CSATA, FACE BATTLE (FaBa).

² A kamera használatánál az arc kivágása alap (lásd pl. a SamuCam projektet) itt felismerés alatt a képek konkrét személyekhez (játékos profilokhoz) vagy jellemzőkhöz (pl. boldog, szomorú) kötéséről van szó.

Fontos, hogy lehetőleg csak arcképeket fotózzunk, ezért az arc fotózós Android app-ban próbáljuk ki az [Android Mobile Vision API](#)-t. A PC-s app-ban kiindulhatunk a [SamuCam](#)-ban használt kódból (itt webkamera van vagy a teló csak egy IP webcam funkcióban és OpenCV-t használtunk).



Fotókkal is műxik

Az említett SamuCam programban láthattuk, hogy az arcfelismerés tanítása fotókkal is tökéletesen (nem a tanulása, a tanítása :) megy, lásd ezen a videón például: <https://youtu.be/6cRbyKr45c>

2.3.1.1.5. Erika

Egy párharcban két család méri össze az erejét (tudását). Mondjuk 5-5 profilképet (saját játékosai) feldobnak a hozzájuk tartozó infókkal, majd ezek rövid tanulása után a másik agynak a család további más képeit kell felismerni, hogy szerepel-e az illető az 5-ben, ha igen, melyik az stb. (kvázi olyan 11-es párbaj jelleggel, illetve az eredményeket itt is a jóslat entrópiája alapján bitekben mérjük).

A győztes család megkapja a legyőzött tudását vagy egy ideig a legyőzött megosztja a legyőzőjével a továbbiakban szerzett információi valamilyen részét.

Az összecsapás elnevezése: agycsata³.

2.3.1.1.6. Konkrét app 5leték

2.3.1.1.6.1. Norbi

A szerveroldal feladata a „Samu az agy” PC-n futó családok és az agycsaták szervezése.

Egy szerver reprezentál egy törzset, a törzsek és a családok kapcsolata 1:N jellegű. Tehát egy törzsnek több családja lehet. A család regisztrációját egy törzshöz a Samu agy kliensen a játékos kezdeményezheti adott esetben.

Tervezz kommunikációs protokollt az alábbi esetekre, használd a [Google Protocol Buffers](#)-t (UDPROG közösségben 6x20 pont, a SamuEntropy forkba ne kerüljön be, bemutatása fészes posztban, a posztodra minden szakmai komment +10 pont).

- i. Norbi és Samu kapcsolata: Norbi minden Samunak annak bitszámától⁴ függő méretű sort biztosít a Gréta, Nándi, Matyi és Erika családtagoknak a képek és az azokat kísérő metaadatok megosztásához.

Tervezz játékosprofil, pl.: kép, név, entrópiája⁵, GPS a regisztrációkor stb. (UDPROG közösségben 35 pont, a SamuEntropy forkba ne kerüljön be, bemutatása fészes posztban, a posztodra minden szakmai komment +10 pont).

- ii. Norbi és Gréta kapcsolata: a Gréta kliens letölti a Samu agy neurális architektúráját, például szerkesztésre.
- iii. Norbi és Nándi kapcsolata: a Nándi kliens letölti a Matyi által „összevadászott” képeket és infókat.
- iv. Norbi és Matyi kapcsolata: a Matyi kliens feltölti az „összevadászott” képeket és infókat.
- v. Norbin keresztül két Samu kapcsolata: az agycsata eredményétől függő információmegosztás a két agy között.
- vi. Norbi és Erika kapcsolata: az agycsata lebonyolítása!
- vii. Két Norbi kapcsolata: törzsek csatája, ezzel egyelőre ne foglalkozunk, majd ha az egy törzsön belüli fejlesztés okés lesz, akkor kerül terítékre!

2.3.1.1.6.2. Samu

Készíts egy arcfelismerő esettanulmányt a TensorFlow (a C++ API) használatával⁶ (UDPROG közösségben 680 pont, a SamuEntropy forkba ne kerüljön be, csak max. saját repóba, bemutatása fészes posztban, a posztodra minden szakmai komment +10 pont).

³ Az elnevezésért köszönet Bátfai Mátyás Bendegúznak.

⁴ A család „jóságát” is bitekben mérjük, azt mondjuk majd, az én családom 64 kilobájtos, 3 gigás stb. jelentsen ez egyelőre bármit.

⁵ Lásd a [Ficsör: bitek gyűjtése](#) c. megjegyzést, ami itt azt fejezné ki, hogy kvázi hány pontos a player.

⁶ Lásd a https://www.tensorflow.org/versions/r0.10/tutorials/image_recognition/index.html tutorialt.

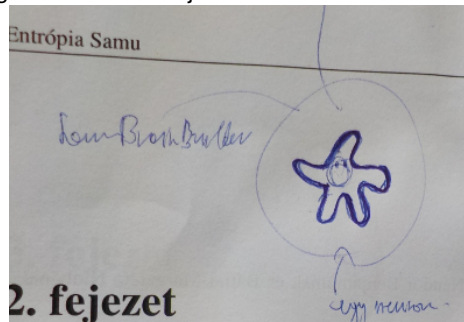
2.3.1.1.6.3. Gréta

Android bevezető feladatként kezdjük az alábbi egymásra épülő apró esettanulmányokkal.

- i. Készíts egy saját forkot a `../cs/NorbironLogo` app-ból (az egész ESAMU-ról persze) és kísérletezz a saját logóddal, hogy mutat! (UDPROG közösségben tipikusan Android kezdőknek ajánlva: 65 pont, a „játékelményt” írd meg fészes posztban is, a posztodra minden szakmai komment +10 pont, minden lájk 5 pont).

Csak tiszta forrásból!

Az ESAMU-ban mindennek tisztának kell lennie szerzői jogi szempontból, sw licenc stb. szempontból! Például a `../res/logo/Norbiron` logó ebből a kézi saját skiccből készült:



- ii. Készíts egy saját forkot a `../cs/NorbironAnim` app-ból (az egész ESAMU-ról persze) és kísérletezz a Norbiron `../res/logo/Norbiron` logó animációjával, hogy mutat! (UDPROG közösségben tipikusan Android kezdőknek ajánlva: 65 pont, a „játékelményt” írd meg fészes posztban is, a posztodra minden szakmai komment +10 pont, minden lájk 5 pont).
- iii. Készíts egy saját forkot a `../cs/NorbironGame` app-ból (az egész ESAMU-ról persze) és kísérletezz a Norbiron `../res/logo/Norbiron` logó animációjával! Tedd például őket dobozokba, ahol az egyik dobozban random mozogjanak, a másikban szabályosan (pl. rácsba szervezve rezegve, vagy körbe forogva stb.) (UDPROG közösségben tipikusan Android kezdőknek ajánlva: 115 pont, a „játékelményt” írd meg fészes posztban is, a posztodra minden szakmai komment +10 pont, minden lájk 5 pont).
- iv. Készíts egy saját forkot a `../cs/NorbironBox` app-ból (az egész ESAMU-ról persze) és kísérletezz a Norbiron `../res/logo/Norbiron` alapú „processzorokkal”! Kapcsold őket például valamilyen hálózatba, ehhez lehet mozgatni kell őket, vagy fix slot helyekre kiválasztani stb. (UDPROG közösségben tipikusan Android kezdőknek ajánlva: 135 pont, a „játékelményt” írd meg fészes posztban is, a posztodra minden szakmai komment +10 pont, minden lájk 5 pont).
- v. Készíts egy saját forkot a `../cs/NorbironBox` app-ból (az egész ESAMU-ról persze) és a [Javadoc](#) konvenciókat követve kommentezd fel a forrásokat! (UDPROG közösségben, tipikusan kezdőknek ajánlva: 90 pont).
- vi. Készíts egy saját forkot a `../cs/NorbironBoard` app-ból (az egész ESAMU-ról persze) és gondold át a processzorok kijelölését, majd valósíts meg ennek kapcsán valamilyen saját elképzelésedet! (UDPROG közösségben, tipikusan kezdőknek ajánlva: 420 pont).
- vii. Készíts egy saját forkot a `../cs/NorbironBoard` app-ból (az egész ESAMU-ról persze) és javíts a mozgatáson és nagyításon (tehát legyen elképzelés mit javítanál és valósítsd is meg)! (UDPROG közösségben, tipikusan kezdőknek ajánlva: 380 pont).
- viii. Készíts egy saját forkot a `../cs/NorbironBoard` app-ból és old meg, hogy a neuron sprite-ok ne másszanak ki a procik alól! (UDPROG közösségben, tipikusan kezdőknek ajánlva: 210 pont).
- ix. Készíts egy saját forkot a `../cs/NorbironBoard` app-ból és old meg, hogy a készülék forgatásakor ua. maradjon az elemek elhelyezkedése! (UDPROG közösségben, tipikusan kezdőknek ajánlva: 310 pont).

A bevezető Androidos feladatokat zárjuk az előző csokor első és utolsó aktivitiének összekapcsolásával, ez a NorbironSplash Készíts egy saját forkot a `../cs/NorbironSplash` app-ból és eszközölj benne valamilyen tetszőleges módosítást! (UDPROG közösségben, tipikusan kezdőknek ajánlva: 80 pont, ennek a hegylakó változata a határidőre, versenyben, from scratch 500 pont volt).

2.3.1.1.6.4. Az 5. védés

Az előző pont oktatásban bevezető (kutatásban a potenciális játékelményeket feltáró) példái után itt azok következnek, amelyek védhetőek a reguláris képzés UDPROG közösségében. A három feladat erőssorrendben követi egymást, ezt is érdemes megfontolni a választásnál.

1. Készíts egy saját forkot a `../cs/NorbironTable` app-ból (az egész ESAMU-ról persze) és a saját magad elképzelte irányban fejleszd tovább, azaz kísérletezz a játékelményvel! Védésként választva kötelező implemntálandó pontok az alábbiak:

- Találj ki valamilyen politikát a nódokat hozzáadó aktiviti elérésére és valósítsd meg (pl. a Build csomópont legyen mindig a bal alsó sarokban, akár legyen külön menü stb. most ugye a Build csomóponton kettőt kattintva jön be).
- Az app emlékezzen a felépített nódokra, azaz újraindítás után a korábban felrakott csomópontok jelenjenek meg.

ezeket készítheted from scratch, vagy akár 2 fős csoportban, a kezdők a további protók alapján, azokra pár napot várva, hogy a from scratch értelmezhető legyen. (UDPROG közösségben 490 pont, a játékelményt írd meg fészes posztban is, a posztodra minden szakmai komment +10 pont, minden lájk 5 pont).

2. Készíts egy saját forkot a `../cs/Brainboard` app-ból (az egész ESAMU-ról persze) és a saját magad elképzelte irányban fejleszd tovább, azaz kísérletezz a játékelményvel! Védésként választva kötelező implemntálandó pontok az alábbiak:

- Tudjon mindent, amit a NorbironTable feladott védési feladatként!
- Az aktuális csomópontokat a BOXIN gomb nyomására tegye egy új dobozba, ami jelenjen meg a BUILD alatt is!

(UDPROG közösségben 790 pont, a játékelményt írd meg fészes posztban is, a posztodra minden szakmai komment +10 pont, minden lájk 5 pont).



Alaplap szerkesztő

A Brainboard app-okban az intuitív cél egyfajta „neurális alaplap” szerkesztőnek mint játéknak a kialakítása.

3. Készíts egy saját forkot a GretaTheBuilder app-ból (az egész ESAMU-ról persze) és a saját magad elképzelte irányban fejleszd tovább, azaz kísérletezz a játékelményvel! Védésként választva kötelező implemntálandó pontok az alábbiak:

- Tudjon mindent, amit a Brainboard feladott védési feladatként!
- Ne csak csomópontokat, hanem éleket is kezeljen vagy mentse/töltse vissza ne csak a csomópontokat, hanem azok pozícióját is az alaplapon.

(UDPROG közösségben 990 pont, a játékelményt írd meg fészes posztban is, a posztodra minden szakmai komment +10 pont, minden lájk 5 pont).

2.3.1.1.6.5. Nándi

2.3.1.1.6.6. Matyi

A korábban említett **SamuCam**-ból kiindulva készíts egy PC-s C++ esettanulmányt, amely az élő képből kijelöli az arcot, lásd még a korábban linkelt SamuCam demó **videót**! (UDPROG közösségben 280 pont, a SamuEntropy forkba ne kerüljön be, csak max. saját repóba, pl. SamuCam fork-ból csak kipusztítani kell, bemutatása fészes posztban).

A korábban említett **Android Mobile Vision API**-t használva készíts egy Androidos esettanulmányt, amely az élő képből kijelöli az arcot. (UDPROG közösségben 440 pont, a SamuEntropy forkba ne kerüljön be, bemutatása fészes posztban, a posztodra minden szakmai komment +10 pont).

2.3.1.1.6.7. Erika

2.3.1.1.6.8. Játékelmény és vizualizációs kísérletek

- i. Samu Amoba: lekérdezi a Norbi szerverről a Samu agy és a család játékosait és az OpenStreetMap-re egy olyan „amóbát” rajzol, ahol az agy GPS koordinátája van középen a nyúlványok a család játékosainak regisztrációkori (vagy vadászatos) GPS koordinátái. (kiindulhatsz a Raxicab [rwin](#) megjelenítőjéből az OSM-re rajzolás tekintetében, a network kommunikációtól tekintünk el, a család GPS koordinátái legyenek behuzalozva a kódba, de persze általánosan működjön, az UDPROG közösségben 420 pont, a SamuEntropy forkba ne kerüljön be, csak max. saját repóba, pl. a Raxicab-ból forkoltba, bemutatása fésztes posztban).

2.3.1.2. Samu OOCWC Fight

Ennek a konkrét játéknak a lényege az OOCWC⁷ RTS kiadása.

2.3.1.3. Samu DevRob Vision

Ennek a konkrét játéknak a lényege a látás kifejlődésének modellezése az 1x1 pixeles szemtől indulva a ???x???-ig.

2.3.1.4. Samu Custom RTS

Ennek a konkrét játéknak a lényege az, hogy a 0 A. D. RTS-ben az agyra bízunk a birodalom adott részeinek irányítását.

2.3.2. Szakfordítói jellegű feladatok

- i. Ennek a doksinak az angol fordítása. (UDPROG közösségben 990 pont)
- ii. Ennek a doksinak az angol fordításának verziókövetése (az új részek fordítása). (UDPROG közösségben 35 pont kommitonként az előző feladat verziójától számolva).
- iii. ...

2.3.3. Licencmérnökség jellegű feladatok

- i. A [Samu Family Album](#) Tensor Flow felhasználásának elemzése, különös tekintettel a „Gréta” szerepre. (UDPROG közösségben nem meghirdetve, csak oktatói forkból)
- ii. A [Samu Family Album](#) „Matyi” szerepének OpenStreetMap felhasználása. (UDPROG közösségben nem meghirdetve, csak oktatói forkból)
- iii. A [Samu Family Album](#) „Matyi” szerepének Google Maps felhasználása. (UDPROG közösségben nem meghirdetve, csak oktatói forkból)
- iv. A [Google Firebase rendszer](#) felhasználásának vizsgálata. Mint bármely más Google szolgáltatásra, erre is az általános [Google APIs Terms of Service](#) vonatkozik. Ami számunkra lényeges, hogy az API-t használó szoftver lehet open source (ez esetben egy fontos megkötés lásd később), de bezárható commercial szoftverként is, ezt a Google nem köti meg. Egyébként jellemzően a Google demo programjai Apache v2 licenccel érhetőek el. Fontos megkötés, hogy ha a szoftver forráskódját open source formában közzétesszük, semmilyen fejlesztői azonosítót nem tehetünk bele (lásd section 4. b. 1.: "Developer credentials may not be embedded in open source projects."). Ezen felül a rendszer használata bizonyos kvóták eléréséig ingyenes, azon felül viszont fizetni kell. Ez nem automatikus, tehát a rendszer nem fog automatikusan fizetési igényt bejelenteni, mert elértünk vagy átléptünk egy kvótát.
[Firebase-t használó gyors prototípus.](#)
- v. Általában a <https://fonts.google.com/> fontok felhasználhatósága és rövid iránymutatás a használathoz. (UDPROG közösségben nem meghirdetve, csak oktatói forkból)

⁷ Az eddigi kiadások a [Police](#) és a [Raxicab](#).

2.3.4. Kreatív és művészeti jellegű feladatok

2.3.4.1. Logók

A logókat a forkban helyezték a docs/res/logo könyvtárba. Kell egy png, egy svg változat és egy licenc! Követendő példaképpen tekintetek a Vona Márton készítette Robotautó Világbajnokság [OOCWC logót](#).

- i. Logó az Entrópia Samuhoz általában. (UDPROG közösségben 240 pont)
- ii. Logó az Entrópia Samu/Samu Family Album játékhöz (Samu + a Gréta, a Nándi, a Matyi és az Erika irányokhoz). (UDPROG közösségben 5x120 pont)
- iii. ...

2.3.4.2. Konkrét app 5letek

- i. A FACE BATTLE mintájára írd le saját játék 5letedet, (UDPROG közösségben 310 pont, például a [Samu OOCWC Fight](#) részben a DocBook forrásban kikommentezett Samu, Gréta, Nándi, Erika, Norbi pontokat kell kitalálnod). A kidolgozás során vedd figyelembe a [Meta-fícsör: egy agy mind felett](#) tippet!

2.3.5. Promóciós jellegű feladatok

- i. Készíts promó YouTube videót az Entrópia Samuhoz. (UDPROG közösségben 540 pont, természetesen csak tiszta szoftverek és erőforrások - képek, fontok, stb. - használhatóak, illetve a gyors protók, feladatmegoldások eredményei, csak az számolhatja el, aki más ESAMU feladatból már elért legalább 150 pontot).
- ii. ...

2.4. Köszönetnyilvánítás

Köszönöm gyermekeimnek Bátfai Mátyás Bendegúznak, Bátfai Nándor Benjáminnak és Bátfai Margaréta Niobénak a remek agyviharokért amelyből kialakult Entrópia Samu koncepciója.

3. fejezet

Tárgymutató

–
0 A. D., 6–8, 18

A

Android Mobile Vision API, 15, 17

B

Brainboard, 17
Budapest 2024, 7, 9

C

Clash of Clans, 8
Bátfai clan, 8

D

DeepMind
TensorFlow, *lásd* Google, *lásd* Google
DevRob, 2
DevRob2Psy, 2, 13
Doom, 7

E

e-Budapest 2024, *lásd* Budapest 2024
ESAMU, 2

G

geocaching, 13
Google
DeepMind, 7, 9
Protocol Buffers, 15
Google Maps, 14, 18
GretaTheBuilder, 17

J

játékélmény, 8, 9, 13, 14
Javadoc, 16
JIBO, 2

K

Killer app, 7

L

League of Legends, 7
LEGO
Mindstorms, 9

M

Minecraft, 8
MIT
média labor, 9

N

Norbiron, 16
NorbironAnim, 16
NorbironBoard, 16
NorbironBox, 16
NorbironGame, 16
NorbironLogo, 16
NorbironSplash, 16
NorbironTable, 17

O

OOCWC, 18, 19
OpenCV, 15
OpenStreetMap, 14, 18
OSM, *lásd* OpenStreetMap

R

Raxicab, 18
rcwin, *lásd* OOCWC
Robotautó Világbajnokság, *lásd* OOCWC
robotpszichológia, 3

S

Samu, 3
Amoba, 18
SRS, 2, 11

T

TensorFlow, 15

U

UDPROG, 3, 13
UDRFT, 13

V

Vona Márton, 19