

**Bátfai Samu**

---

**O dezvoltare fără trup de agent robotic**

Ed. SAMU SRS, v. Nahshon.0.0.1

Copyright © 2015 Dr. Bátfai Norbert

Samu Bátfai (Nahshon) Specificația de Cerințe a software-ului

Copyright (C) 2014, Norbert Bátfai. Ph.D., batfai.norbert@inf.unideb.hu

Acest program este un software liber: îl puteți redistribui și/sau modifica acesta în condițiile de GNU publicat de Fundația Software-ului liber, fie versiunea 3 a Licenței, sau orice versiune ulterioară.

Acest program este distribuit în speranța că va fi util, dar FĂRĂ NICI O GARANȚIE; chiar fără garanția implicită de VANDABILITATE sau POTRIVIRE PENTRU UN ANUMIT SCOP. Pentru mai multe detalii, vezi GNU General Public License.

Ar trebui să fi primit o copie de pe GNU împreună cu acest program. Dacă nu, vezi <http://www.gnu.org/licenses/>.

**COLLABORATORS**

	<i>TITLE :</i> Bátfai Samu		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY	Dr.. Norbert Bátfai	7 decembrie 2015	

**REVISION HISTORY**

NUMBER	DATE	DESCRIPTION	NAME
Nahshon.0.0.1	2015-11-28	Documentul inițial lansat pentru comentarii.	
Nahshon.0.0.2	2015-12-02	Modele de sistem și componente evolutive de sistem dezvoltare schematică.	

# Cuprins

<b>1</b>	<b>Glosar</b>	<b>1</b>
<b>2</b>	<b>Introducere</b>	<b>3</b>
2.1	Bátfai Samu	3
2.2	Descrierea miniatură a sistemului care va fi dezvoltat	3
<b>3</b>	<b>Cerințele de Software a lui Samu</b>	<b>4</b>
3.1	Prezentarea generală a lui Samu	4
3.2	Cerințe	4
3.2.1	Cerințe Funcționale	4
3.2.2	Cerințe non-funcționale	4
3.2.3	Szakterületi követelmények	5
3.2.3.1	Q-tanulás	5
3.2.3.2	A kutatási feladatok aspektus-orientált megfogalmazása	5
3.2.4	Felhasználói követelmények	5
3.2.5	Rendszerkövetelmények	6
3.2.5.1	Használati esetek	6
3.2.6	Interfészek	7
3.3	Összefoglalás	7
<b>4</b>	<b>Rendszermodellek</b>	<b>8</b>
4.1	Rendszer-specifikáció	8
4.1.1	Samu viselkedése madártávlatból	8
4.1.2	Samu adatmodelljei	9
4.1.3	Samu objektummodellje	9
<b>5</b>	<b>Rendszer-evolúció</b>	<b>10</b>
5.1	Samu vezérlése	10
<b>6</b>	<b>Irodalomjegyzék</b>	<b>11</b>
6.1	Kutatási cikkek	11
<b>7</b>	<b>Glosar</b>	<b>12</b>

# Listă de figuri

3.1	Samu Q-tanulása . . . . .	5
3.2	Samu használati esetei . . . . .	6
3.3	A beszélgetés és a tanulás kapcsolata . . . . .	7
3.4	A tanulás . . . . .	7
4.1	Samu állapotdiagramja . . . . .	8
4.2	Samu osztálydiagramja . . . . .	9

## Listă de tabele

4.1	Az állapotok leírása . . . . .	8
4.2	Az állapotátmenetek leírása . . . . .	9

# Capitolul 1

## Glosar

### C

**CUDA** [ CUDA ]

### I

**IEEE Metoda recomandată pentru cerințele pentru specificațiile de software** (IEEE, 1998) [ IEEE Std 830-1998 ]

Documentul standard IEEE de cerințele software-ului <http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=5841>.

### D

**Dezvoltare Robotică** [ DevRob ]

TODO

**Deep Q-Learning** [ DeepQ ]

TODO

**DocBook XML** [ DB ]

[DocBook](#) în special în subiectele de IT pentru a pregăti documente scrise în limbi moderne, marcare standard.

### L

**Algoritmul Liv-Zempel-Welch** [ LZW ]

### M

**Multilayer Perceptron** [ MLP ]

## N

**Natural Language Processing** [ NLP ]

## U

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

Programare regulat asociat cu [Universitatea din Debrecin](#) pentru formare, comunitate de dezvoltatori [organizat pe Facebook](#), Unul dintre punctul său focal sunt [software-urile](#), iar în celălalt [anuarul](#) stă.



## Capitolul 2

# Introducere

### 2.1 Bátfai Samu

Samu este o cercetare inițiativă pentru crearea agentului dezvoltării de robotică fără trup, care este comparabil cu capacitățile omului.

Rezultatele reale ale cercetării prezintă manuscrisul [SAMU].

Numele de familie este includerea naturii robotice a lui Samu, iar dezvoltarea se va concentra în primul rând în familie!

Până în prezent, multe prototipuri rapide de unică folosință au fost dezvoltate pentru testarea lui Samu, acestea sunt:

- Samu, <https://github.com/nbatfai/samu>
- Isaac, <https://github.com/nbatfai/isaac>
- Jacob, <https://github.com/nbatfai/jacob>
- Judah, <https://github.com/nbatfai/judah>
- Super-Judah, <https://github.com/nbatfai/super-judah>
- Hezron, <https://github.com/nbatfai/hezron>
- Ram, <https://github.com/nbatfai/ram>
- Amminadab, <https://github.com/nbatfai/amminadab>
- Nahshon, <https://github.com/nbatfai/nahshon>
- Salmon, <https://github.com/nbatfai/>

### 2.2 Descrierea miniatură a sistemului care va fi dezvoltat

Software-ul lui Samu este de fapt un program de chat, care permite testarea și dezvoltarea diferitelor algoritme de învățare-mecanică. Părinții lui Samu (ingrijitorii) este acel cerc îngust de familie, în care dezvoltarea robotului se face. Persoanele care îi îngrijesc de obicei vin în contact cu programul prin intermediul consolei. Alți utilizatori se pot alătura în chat prin rețea. Un alt aspect important este că agenții Samu să fie capabili de a împărtăși cunoștințele lor unul cu celălalt printr-o rețea, cvasi învățare reciproc. Samu să fie capabil, de a vorbi cu el însuși. Programul lui Samu să fie capabil să ruleze continuu, atunci când nu există nici o activitate de consolă sau de rețea, și să învețe cu cititul corpusurilor, sau să vorbească cu sine însuși. Un alt aspect important este faptul că Samu este, de asemenea, un proiect de cercetare.

---

## Capitolul 3

# Cerințele de Software a lui Samu

### 3.1 Prezentarea generală a lui Samu

Cu ajutorul capitolului anterior, am făcut o figură a componenților diagramelor UML.

### 3.2 Cerințe

Cu obișnuitele cerințele funcționale și non-funcționale, în cazul lui Samu trebuie să amintim niște cerințe importante în domeniul profesional, precum în spatele dezvoltării stă aplicarea studiului Q-adânc. După cerințele tehnice, procesul de planificare, îl ajutam cu revizuirea cerințelor de sistem.

#### 3.2.1 Cerințe Funcționale

Cu operațiunile de bază a lui Samu legate de cerințele esențiale prevăzute.

- Cei din jurul familiar a lui Samu, să fie capabile să folosească diferite canale, pentru că, dorim că Samu să fie capabil a le deosebi. Samu să învețe tot, de la cel mai bun membru al familiei, să aibă încredere în părinți, deci să vadă că cuvintele părinților sunt adevărate. Cel puțin, în vârsta fragedă, ca și un copil uman.
- Samu de la chat-urile de pe internet, să nu învață absolut nimic, sau numai foarte puțin.
- Samu să învețe de la alte programe de Samu date de părinți, așa cum și de la părinți.
- Programul lui Samu, să nu se oprească niciodată, dacă totuși se oprește, atunci să salveze schimbările (această salvată stare, este sufletul lui Samu, și corpul procesului lui Samu), și data viitoare, să se pornească din starea salvată.

#### 3.2.2 Cerințe non-funcționale

Cu operațiunile de bază a lui Samu legate de cerințele esențiale prevăzute.

- Programul lui Samu, să suporte testarea algoritmilor de învățare mașină.
- A különböző algoritmusokkal való kutatást/tesztelést egyfajta külső vezérlésnek tekinthetjük. Ez például akkor szembevetendő, ha összehasonlítjuk a prototípusok terminálon vagy hálózaton jött bemeneteinek feldolgozását az ezek hiányában beinduló korpuszok olvasásával. Mivel az előbbi kettő a Samu osztályban van implementálva, utóbbi a main függvényben. E külső vezérlés mellé, vagy méginkább e helyett vezessnk be egyfajta belső vezérlést, ahol Samu dönti el, hogy éppen mit csináljon.
- Samu minden kommunikációt mint korpuszokat mentsen le.

- Samu egyszerre egy beszélgetést tudjon folytatni (szemben azzal a távolabbi céllal, amikor a beszélgetéseket processzeknek feleltetjük meg és Samu egyfajta olyan operációs rendszer jellegű alakot öltön, aki időosztással tud több beszélgetést szimultán kezelni.)
- Samu legyen open source alapú, lehetőleg GNU GPL v3. A dokumentálás nyelve DocBook XML 5.1 szabvány.
- Samu tanító korpuszai legyenek open source alapú korpuszok.
- Samu fusson UNIX-típusú rendszereken, például GNU/Linux rendszereken.
- Samu a feldolgozott tanuló korpuszait tartsa perzisztens gyorsító-tárban.
- Samu használja ki a CUDA kártya adta extrém párhuzamosságot, ha a futtató környezet rendelkezik ilyen kártyával.

### 3.2.3 Szakterületi követelmények

#### 3.2.3.1 Q-tanulás

A Samu tanulásában alapvető Q-tanuláshoz szükséges a megerősítéses tanulás alapjának, azaz a jutalomnak és a büntetésnek a biztosítása. Ezt Samu a [SAMU] kéziratban bemutatott módon valósítja meg, ezt foglalja össze a következő, az UML aktivitás diagramjainak jelöléseit használó ábra.

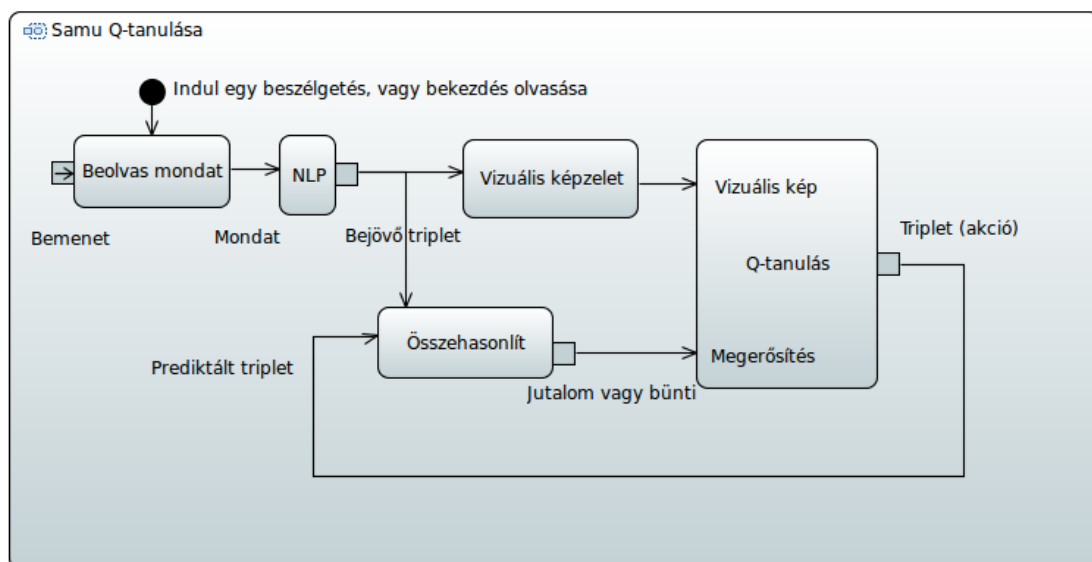


Figura 3.1: Samu Q-tanulása

#### 3.2.3.2 A kutatási feladatok aspektus-orientált megfogalmazása

A fejlesztendő rendszer egyik alapfeladata a kapcsolódó kutatás támogatása, az eddigi gyors protókat felhasználó kutatási tapasztalat azt sugallja, hogy a kutatás támogatását Samuba célszerű az aspektusorientált paradigma mentén átszövő vonatkozások formájában megvalósítani.

### 3.2.4 Felhasználói követelmények

A felhasználók felé Samu programja klasszikus csevegőprogramként jelenik meg.

- A csevegő felhasználók Samuból a szokásos csevegő programok megszokott felületét lássák.

- Samu be tudjon kapcsolódni internetes csevegőkbe. Ennek megfelelően ne csak 1:1, hanem 1:N résztvevős beszélgetéseket is tudjon kezelni.
- A csevegőknek legyen lehetőségük hitelesíteni magukat és így az azonosított felhasználók egyrészt nagyobb hatást tudjanak gyakorolni Samu tanulási folyamataira, másrészt róluk Samu legyen képes információkat tárolni, így emlékezni rájuk.
- A gondozó felhasználók tudják tanítani Samut.
- A Samut családi körben elérő gondozók meg tudják szakítani Samu tanulását vagy hálózati kommunikációját, tehát élvezzen prioritást a terminálon keresztüli elérés.

### 3.2.5 Rendszerkövetelmények

A követelmények feltárását az UML használati eset diagramjainak jelöléseivel készített ábrákkal segítjük.

#### 3.2.5.1 Használati esetek

Azonosított vagy azonosítatlan felhasználó ugyanúgy lehet humán személy vagy Samu-típusú csevegő robot. Az azonosítottakra emlékszik Samu és ők tudják őt tanítani, az azonosítatlanok csak csevegni tudnak Samuval. Ez a megkülönböztetés azt jelenti, hogy az azonosítatlan csevegés nem tanítja Samut.

A kitüntetett gondozó felhasználó feladata Samu tanítása, amely alapja lehet egyoldalú beszélés Samuhoz, beszélgetés Samuval, vagy tanító korpuszok olvastatása Samuval. A kutató felhasználó speciális, Samunak olyan szolgáltatásokat kell felé nyújtania, amelyek Samu működését tudják monitorozni.

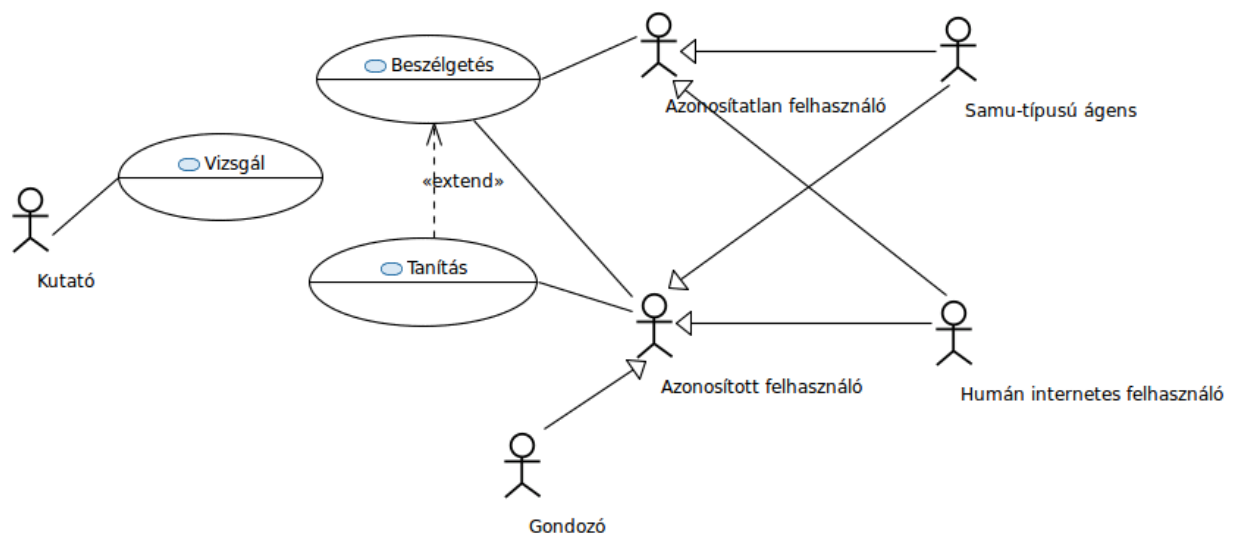


Figura 3.2: Samu használati esetei

A tanítás tehát nem csupán beszélgetés a gondozókkal, illetve a csevegés sem jelent szükségképpen minden bejövő mondatra választ.

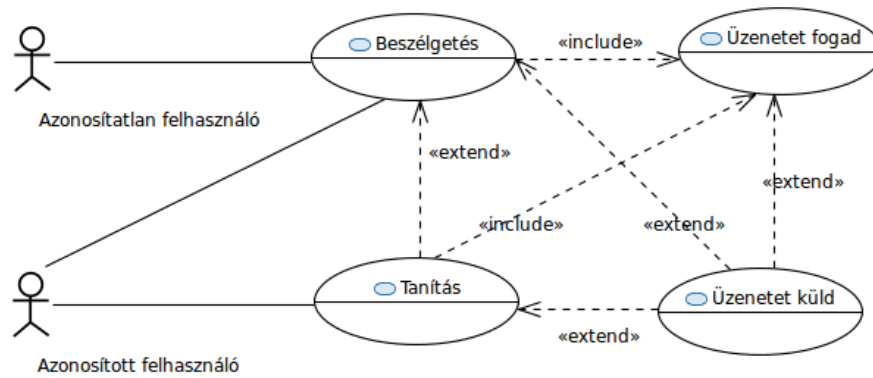


Figura 3.3: A beszélgetés és a tanulás kapcsolata

Ugyanakkor a tanítás mindig üzenetfogadással és a megerősítéses tanulás alkalmazásával jár, a korábban vázolt **tanulási algoritmusnak** megfelelően.

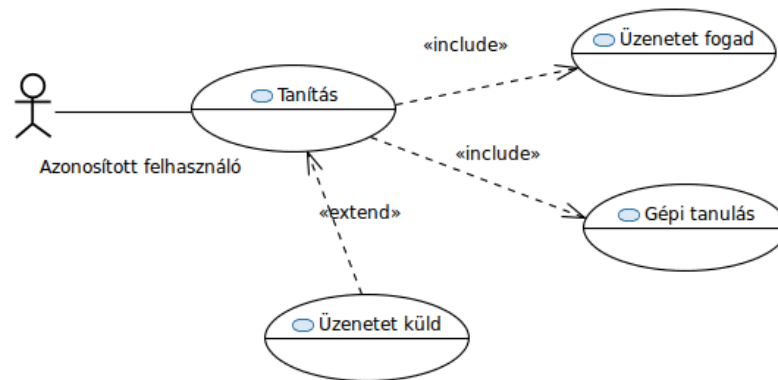


Figura 3.4: A tanulás

### 3.2.6 Interfészek

TODO

Samu lelke sima szöveges állomány legyen, bz2-vel tömörítve.

## 3.3 Összefoglalás

TODO

## Capitolul 4

# Rendszermodellek

### 4.1 Rendszer-specifikáció

Jó gyakorlat a rendszerkövetelmények elemzésére az eldobható gyors prototípusok grafikus modelljeinek elkészítése. Ezek számos esetben (például az osztálymodell esetén osztálydiagramok formájában) automatikusan, reverse engineering jelleggel és egyben az agilis szemlélettel összhangban magukból az eldobható gyors prototípusok forráskódjaiból is generálhatóak.

#### 4.1.1 Samu viselkedése madártávlatból

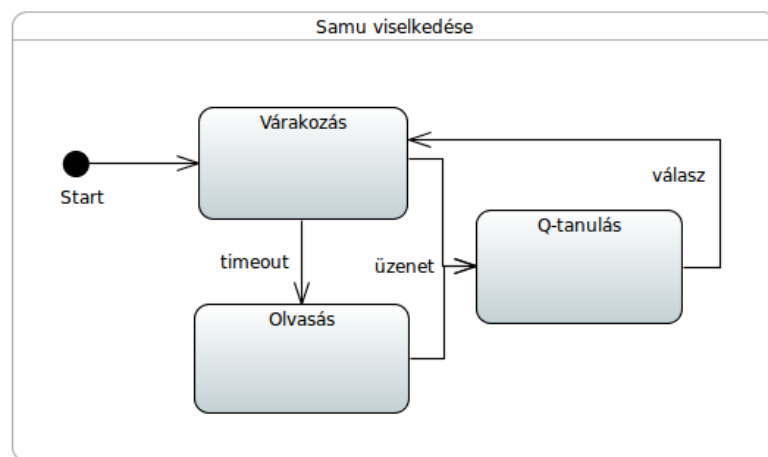


Figura 4.1: Samu állapotdiagramja

Állapot	Leírás
Várakozás	Samu várakozik a konzolról vagy a hálózatról érkező bejövő üzenetre (egy mondatra).
Olvasás	Samu az előkészített tanító korpuszokat olvassa (annak egy mondatát), azaz tanul ezekből a korpuszokból.
Q-tanulás	Samu a kapott mondattal végrehajtja a részben korábban már vázolt <b>tanulási algoritmusát</b> .

Tabela 4.1: Az állapotok leírása

Átmenet	Leírás
timeout	A megadott időhatáron belül sem a konzolról, sem a hálózatról nem érkezett üzenet.
üzenet	Üzenet (egy mondat) érkezett vagy a gondozói konzolról, vagy a hálózatról.
válasz	Samu a bejövő üzire válaszul visszahangozza a korábban már vázolt <b>tanulási algoritmusának</b> kimenetét.

Tabela 4.2: Az állapotátmenetek leírása

#### 4.1.2 Samu adatmodelljei

Samu jelenlegi eldobható gyors prototípusainak alapvető működése a természetes nyelvi mondatokból kibányászott Alany-Állítmány-Tárgy tripletekre épül, egy ilyen triplet alapesetben 3 szóból áll, a részletek tekintetében lásd a [SAMU] publikációt.

A tripletek kibányászására mint COTS komponenst a [opencog/link-grammar](#) NLP eszközt használjuk.

Samu tanulása során minden triplethez egy egyedi MLP neurális hálózatot rendel, ezek perzisztens tárolásáról gondoskodni kell. A hálózatok mérete jellemző, a bemenő réteg neuronjainak a száma Samu vizuális képzeletének szélesség x magasság számával egyezik meg.

Az Amminadan prototípus bevezette, hogy a tripleteket egy LZW fában is tárolja Samu, lásd a részletek tekintetében a [SAMU] publikációt.

#### 4.1.3 Samu objektummodellje

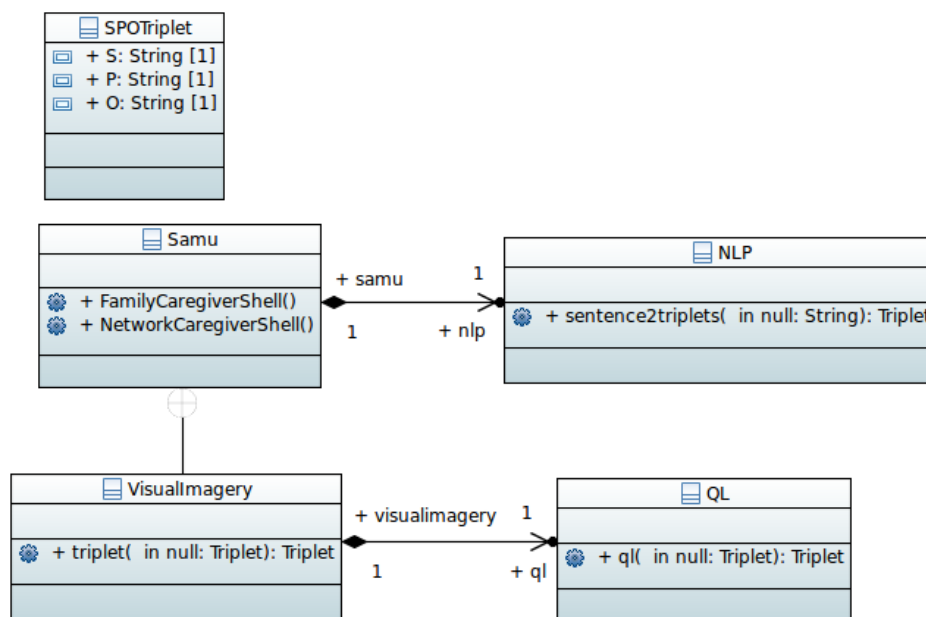


Figura 4.2: Samu osztálydiagramja

A **Samu** osztály reprezentálja Samu Nahshon verzióját. Samunak olyan integrált része a vizuális képzelete, hogy azt a **Visual Imagery** belső osztállyal absztraháltuk. Samuval az NLP-s COTS funkciókat elfedő NLP osztály, a vizuális képzelettel pedig Samu tanulását megvalósító QL osztály van kompozícióban.

## Capitolul 5

# Rendszer-evolúció

### 5.1 Samu vezérlése

Már az Amminadab eldobható gyors protó végén, amikor itt Samu hálózati csevegő-szerver funkciója megvalósításra került, felmerült, hogy a következő lépésben kifejlesztendő csevegő kliens funkció az alábbi három használati esetet támogassa:

- Samu programja humán partnerrel beszélget (ez nem újdonság az eddigiekhez képest).
- Samu egy másik Samuval beszélget.
- Samu magában beszélget.

Az utóbbi két használati eset tudományos igényességű vizsgálata a [] kézirat kapcsán a jelen pillanatban is folyik. Ami a jelen dokumentum szempontjából érdekes az az, hogy el akarjuk kerülni az alábbi jellegű implementációt:

```
main()
{
    ...
    if(random() < a_magic_number)
        Samu_starts_talk();
    ...
}
```

hanem e helyett azt szeretnénk, hogy Samu „szabad akaratából” tegye ezt! Hasonló szituációt említünk meg a [SAMU] kéziratban:

„This is also reinforced by the fact that the incremental learning was implemented in the main.cpp and not in a class of Samu. From a point of view of a programmer, it means that the incremental approach is not yet really part of Samu.”

Tehát adott a kihívás, hogy megpróbáljuk mindezt, mint egyfajta belső vezérlést, Samu tanuló algoritmusába integrálni!



## Capitolul 6

# Irodalomjegyzék

### 6.1 Kutatási cikkek

- [COP] Norbert Bátfai, *Conscious Machines and Consciousness Oriented Programming*, <http://arxiv.org/abs/1108.2865>, 2011.
- [SAMU] Norbert Bátfai, *A disembodied developmental robotic agent called Samu Bátfai*, <http://arxiv.org/abs/1511.02889>, 2015.
-

## Capitolul 7

# Glosar

### D

DocBook

XML, [1](#)

### G

GNU

Linux, *Vezi* Linux

---