

Bátfai Samu

O dezvoltare fără trup de agent robotic

Ed. SAMU SRS, v. Nahshon.0.0.3

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	14 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.	
Nahshon.0.0.3	2015-12-04	Ridicarea lui SAMU-OOCWC și reparații.	

Cuprins

1	Glosar	1
2	Introducere	3
2.1	Bátfai Samu	3
2.2	Descrierea miniatură a sistemului care va fi dezvoltat	3
3	Cerințele de Software a lui Samu	4
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	Cerințe de Specialitate	5
3.2.3.1	Învățare-Q	5
3.2.3.2	Formulare orientate aspect al sarcinilor de cercetare	5
3.2.4	Cerințe de Utilizatori	5
3.2.5	Cerințe de sistem	6
3.2.5.1	Cazuri de utilizare	6
3.2.6	Interfațe	7
4	Modelle de Sistem	8
4.1	Specificație de Sistem	8
4.1.1	Comportarea lui Samu	8
4.1.2	Modelele de date a lui Samu	9
4.1.3	Modelul de obiect a lui Samu	9
5	Referințe	10
5.1	Articole de cercetare	10
6	Glosar	11

Listă de figuri

3.1	Învăţarea-Q a lui Samu	5
3.2	Cazuri de utilizare a lui Samu	6
3.3	Relaţia dintre învăţare şi conversaţie	6
3.4	Învăţarea	7
4.1	Diagramul de stare a lui Samu	8
4.2	Diagramul de clasă a lui Samu	9

Listă de tabele

4.1	Descrierea stărilor	8
4.2	Descrierea tranzițiilor de stări	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 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 alogoritme 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.
 - Cercetările și testările cu deosebitele algoritme, putem considera ca un fel de control extern. De exemplu acest lucru atunci este izbitoare, când comparăm prelucrarea inputelor venite de pe rețea sau de pe terminal a prototipelor cu pornirea citirii a corpușilor în lipsa ei. Deoarece cele precedente sunt implementate în clasa lui Samu, cel din urmă în funcția main. Lângă acest control extern, sau chiar în locul acestuia să introducem un control intern, unde Samu consideră, ce să facă.
 - Samu să salveze toate comunicațiunile ca și corpusuri.
-

- Samu în același timp să poată continua numai o conversație (față co scopul, când conversațiile sunt niște procese și Samu un fel de sistem de operare.)
- Samu să fie un software liber, tip de GNU GPL v3. Limba documentală este norma DocBook XML 5.1
- Corpusele învățătoare a lui Samu să fie tip soft liber.
- Samu să funcționeze pe sisteme de UNIX, de exemplu în sistemele:

3.2.3 Cerințe de Specialitate

3.2.3.1 Învățare-Q

Samu să primească recompensă sau pedeapsă. O figură care reprezintă acest lucru, este atașat mai jos.

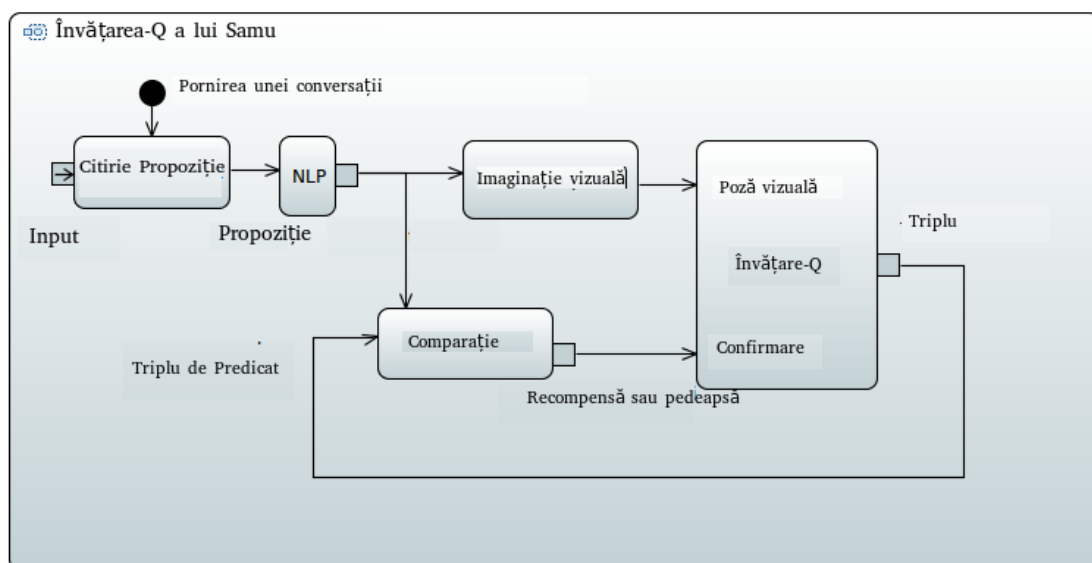


Figura 3.1: Învățarea-Q a lui Samu

3.2.3.2 Formulare orientate aspect al sarcinilor de cercetare

Una dintre principalele sarcini ale sistemului dezvoltat de cercetare necesare pentru a sprijini existente prototipe de cercetare experiență de utilizator rapid arată că sprijinul pentru cercetare de SAMU ar trebui să constituie de-a lungul paradigmei predominantă în aspecte pentru realizare.

3.2.4 Cerințe de Utilizatori

Samu pentru utilizatori apare ca un program classic de conversații.

- Utilizatorii din Samu, să vadă o suprafață classică de conversații.
- Samu să poată conecta în chat-uri pe internet. Prin urmare să poată să discute cu mai multe persoane în același timp.
- Utilizatorii să poată învăța pe Șamu.
- Controlul prin consol să fie în prioritate față cu cel de pe rețea (internet).

3.2.5 Cerințe de sistem

Cerințele pentru divulgarea de utilizare UML caz diagrame marcaje făcute diagrame ajutatoare.

3.2.5.1 Cazuri de utilizare

Utilizatorul poate fi identificat sau nu este identificat indivizi umani sau aceleași tip Samu de chat robot. Samu își amintește de cele identificate și este în măsură să-l învețe, neidentificatele poate discuta doar cu Samu.

Utilizatorul grija deosebită învățătura lui Samu, care se poate baza pe o singură față Samu vorbesc, conversație cu Samu, un profesor sau citind corpusuri. Cercetătorul trebuie să ofere utilizatorilor specifice, servicii Samu, care sunt Operație Samu pot fi monitorizate.

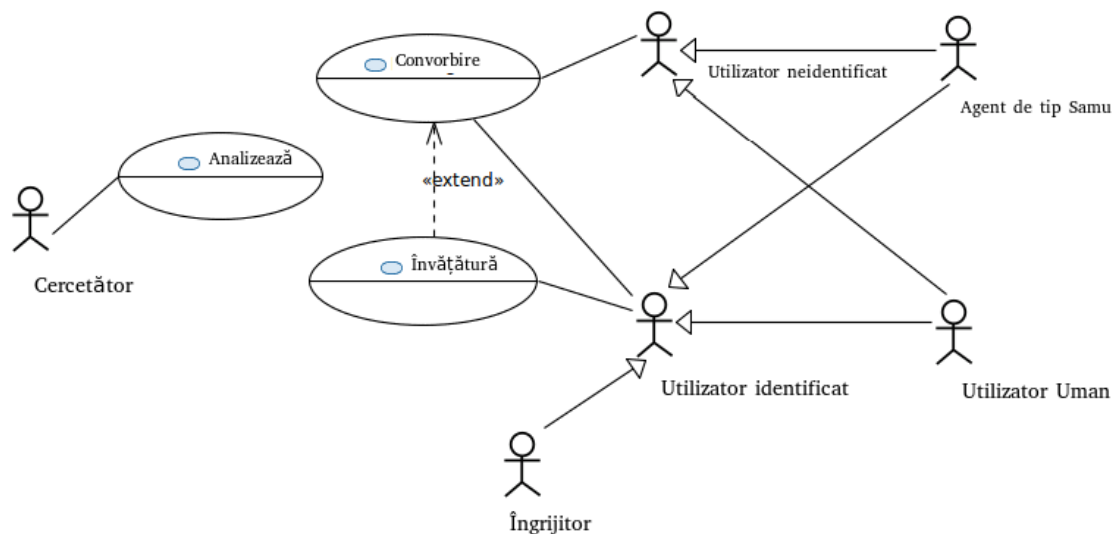


Figura 3.2: Cazuri de utilizare a lui Samu

Învățarea nu înseamnă convorbirea cu utilizatorul.

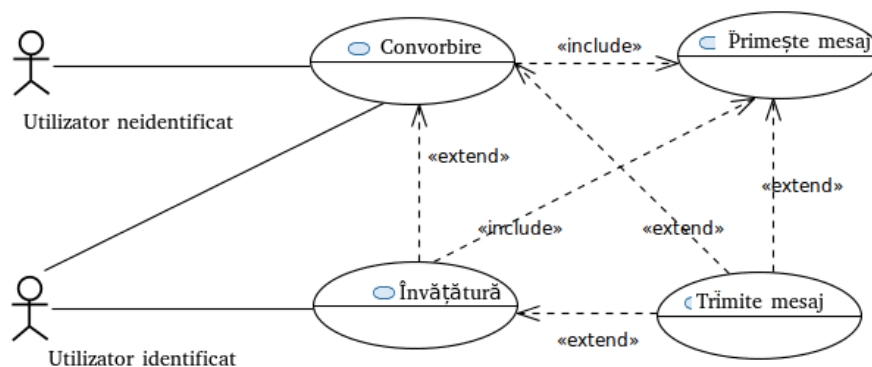


Figura 3.3: Relația dintre învățare și conversație

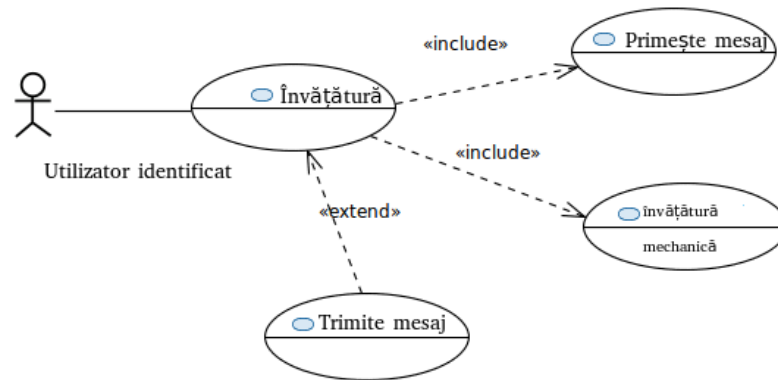


Figura 3.4: Învățarea

3.2.6 Interfațe

Sufletul lui Samu să fie doar text, archivat cu bz2.

Capitolul 4

Modelle de Sistem

4.1 Specificație de Sistem

Este o bună practică de a analiza cerințele de sistem pentru unică folosință rapid elaborarea de modele grafice prototipuri. Acestea sunt, în multe cazuri, (de exemplu, modelul de clasă după caz, de clasă) în mod automat, inginerie inversă, și, de asemenea, caracterul agil prototipuri rapide de unică folosință în sine forráskódjaiból este în concordanță cu abordarea poate fi generat.

4.1.1 Comportarea lui Samu

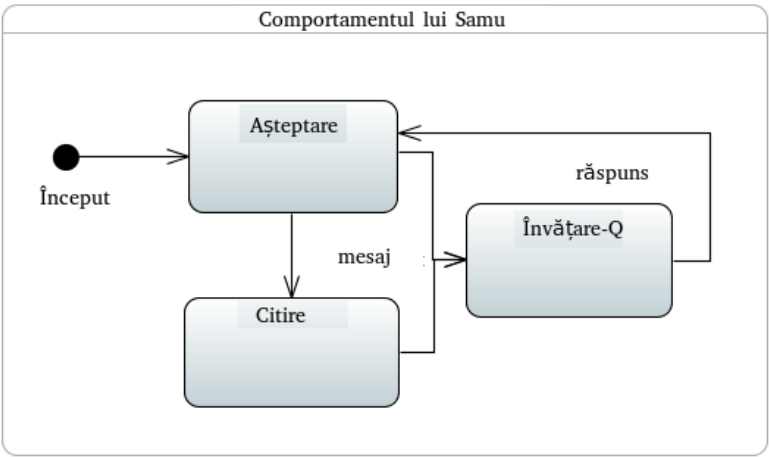


Figura 4.1: Diagramul de stare a lui Samu

Stare	Descriere
Așteptare	
Citire	
Învățare-Q	

Tabela 4.1: Descrierea stărilor

Tranziție	Descriere
Lipsă de timp	În termenul specificat, fie consola sau rețeaua nu primește mesaj.
Mesaj	Mesaj (o propoziție) de pe consol sau rețea.
Răspuns	

Tabela 4.2: Descrierea tranzițiilor de stări

4.1.2 Modelele de date a lui Samu

Funcționarea de bază de curent prototip rapid disponibil Samu se bazează pe propoziții limbaj natural minat triple subiect-obiect-predicate, triple astfel de default sunt de trei cuvinte.

Pentru citirea tripletelor folosim NLP [opencog/link-grammar](#)

4.1.3 Modelul de obiect a lui Samu

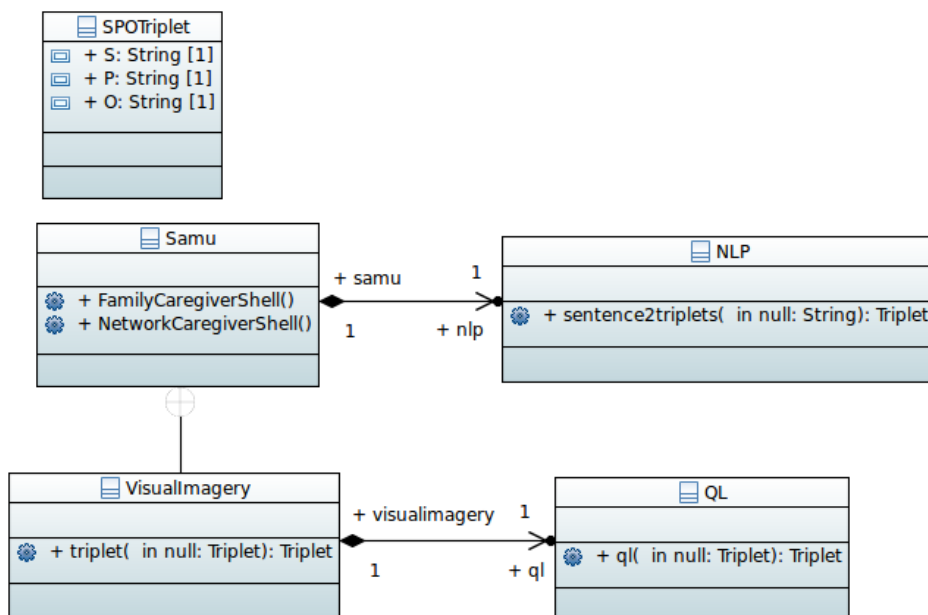


Figura 4.2: Diagramul de clasă a lui Samu

Clasa Samu reprezintă versiunea Nahson din Aleșd. Samu are încă și ImaginareVizuală de magnat. Samu are o clasă numită NLP osztály, și o altă numită QL.

Capitolul 5

Referințe

5.1 Articole de cercetare

- [COP] Norbert Bátfai, *Conscious Machines and Consciousness Oriented Programming*, <http://arxiv.org/abs/1108.2865>, 2011.
- [SAMU] Norbert Bátfai, *O dezvoltare fără trup de agent robotic numit Bátfai Samu*, <http://arxiv.org/abs/1511.02889>, 2015.

Capitolul 6

Glosar

D

DocBook

XML, [1](#)

G

GNU

Linux, *Vezi* Linux
