

Team : Discount Siri

Members :

- Gheorghe Adrian-Valentin
- Cutcunel Dragos-Ioan
- Damsa Elena-Veronica
- Popescu Ioana
- Neamtu George-Nicolae
- Ene Andrei

Specificația cerințelor

Introducere

· ***Scurta descriere a produsului***

Aplicatia noastra reprezinta un laborant vocal. Modul lui de functionare are urmatorul sistem : de la tastatura laborantul primeste o comanda, acesta o interpreteaza si ofera un raspuns in functie de comanda primita. Laborantul poate oferi raspunsuri la comenzi de sistem, poate deschide diferite programe si poate intoarce un rezultat pentru functii care folosesc internetul. Pentru anumite functii, intoarce rezultate si pentru un subset de date.

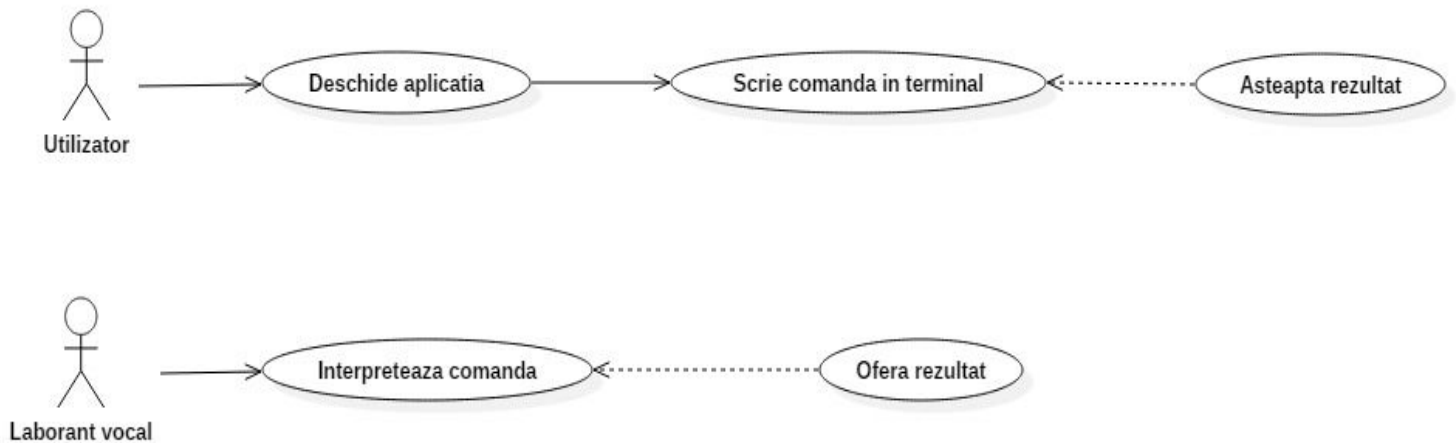
Descrierea cerințelor

· ***Cerințe de sistem (echipamente, comunicație, etc.)***

Cerintele minime de sistem ce trebuiesc respectate pentru rularea optima a programului sunt:

- Sistem de operare Windows 7;
- 4Gb memorie RAM;
- 1Gb memorie grafica (integrata sau dedicata);
- Procesor Intel Quad Core 2GHz;
- Mouse si Tastatura.

· **Diagrama de context (cu toate cazurile de utilizare)**



· **Descrierea fiecărui caz de utilizare**

Cazul de utilizare „Deschide aplicatia”:

- Actor/i implicați : Utilizator
- Prioritatea (critică/ridicată/medie/scazută) : ridicata
- Fluxul de baza :
 - a) Precondiție (starea sistemului necesară pentru începerea cazului de utilizare)

Un utilizator trebuie sa porneasca aplicatia pentru a comunica cu laborantul vocal.
 - b) Pași (secvența tipică de execuție)

-

Cazul de utilizare „Scrie comanda in terminal”:

- Actor/i implicați : Utilizator
- Prioritatea (critică/ridicată/medie/scazută) : ridicata
- Fluxul de baza :
 - a) Precondiție (starea sistemului necesară pentru începerea cazului de utilizare)

Un utilizator trebuie sa porneasca aplicatia pentru a comunica cu laborantul vocal.
 - b) Pași (secvența tipică de execuție)

Dupa ce aplicatia a fost pornita, utilizatorul poate scrie o comanda in terminal pentru a fi preluata de laborant.

Cazul de utilizare „Asteapta rezultat”:

- Actor/i implicați : Utilizator
- Prioritatea (critică/ridicată/medie/scazută) : ridicata

- Fluxul de baza :

a) Precondiție (starea sistemului necesară pentru începerea cazului de utilizare)

Un utilizator trebuie sa porneasca aplicatia pentru a comunica cu laborantul vocal.

b) Pași (secvența tipică de execuție)

Dupa ce utilizatorul a scris comanda in terminal, el asteapta raspunsul laborantului vocal.

Cazul de utilizare „Interpreteaza comanda”:

- Actor/i implicați : Laborant

- Prioritatea (critică/ridicăta/medie/scazută) : ridicata

- Fluxul de baza :

a) Precondiție (starea sistemului necesară pentru începerea cazului de utilizare)

Un utilizator trebuie sa porneasca aplicatia pentru a comunica cu laborantul vocal si sa scrie o comanda in terminal.

b) Pași (secvența tipică de execuție)

Dupa ce utilizatorul a scris comanda in terminal, laborantul preia comanda pentru a o interpreta. Comanda primita o va parsa pentru a identifica tipul comenzii si daca comanda are parametrii. Utilizatorul vocal poate sa interpreteze urmatoarele comenzi :

Nr.crt.	Comenzi de sistem
1.	time [attr]
2.	date
3.	ip
4.	del [filename]
5.	mkdir [directoryname]
6.	shutdown

Nr.crt.	Programe
1.	notepad
2.	cmd
3.	player
4.	calculator

5.	paint
----	-------

Nr.crt.	Funcții care presupun utilizarea internetului
1.	mail
2.	twitch + [gameName]
3.	weather [attr]
4.	currency [attr]

Unele dintre comenzi pot avea și un subset de opțiuni :

Aplicatie	Opțiuni
1. weather [attr]	*
2. time [attr]	Boston
	Bogota
	Berlin
	Tokyo
3. currency [attr]	eur, usd, gbp, inr etc.

Cazul de utilizare „Oferta rezultat”:

- Actor/i implicați :
- Prioritatea (critică/ridicăta/medie/scazută) : ridicata
- Fluxul de baza :
 - a) Precondiție (starea sistemului necesară pentru începerea cazului de utilizare)

Un utilizator trebuie să pornească aplicația pentru a comunica cu laborantul vocal și să scrie o comandă în terminal.
 - b) Pași (secvența tipică de execuție)

După ce a interpretat comanda, laborantul oferă rezultatul vocal pentru comanda introdusă de utilizator.

Limbaje de programare

Laborantul vocal este implementat în Java, folosind IDE-ul NetBeans, iar repository-ul aferent aplicației este ținut pe platforma Github.

Logica aplicației

Utilizatorul deschide aplicația și scrie comanda în terminal. Comanda este preluată și parsată pentru a se identifica tipul și dacă aceasta are parametri. Dacă utilizatorul introduce o comandă de sistem, aceasta este realizată cu ajutorul funcțiilor din Java și apoi rezultatul este întors vocal.

Dacă se dorește deschiderea unui anumit program, utilizatorul poate scrie în terminal numele programului. Acesta este preluat de laborant, se identifică exe-ul programului și se deschide. Rezultatul comenzii este întors vocal. Închiderea programului se face prin rescrierea numelui în terminal.

Dacă utilizatorul introduce o comandă care necesită conectarea la internet, se stabilește conexiunea la internet, se deschide o pagină în browser în funcție de comandă introdusă și parametri, iar rezultatele sunt întoarse în format vocal de către laborantul vocal (mesaj de succes).