

MansInventive



PER ASPERA AD ASTRA

Echipa responsabilă de realizarea proiectului:

- ◆ Sd. Cap. Bordei Alin-Viorel
- ◆ Sd. Cap. Coșerea Ionuț



Numărul versiunii	Modificări adăugate
Versiune1	Versiune inițială
Versiune2	Actualizarea cerințelor funcționale și nefuncționale

Cuprins

Capitolul 1	3
Scopul proiectului.....	3
Lista definițiilor	3
Structura DCS.....	4
Capitolul 2	4
Descrierea produsului software	4
Detalierea platformei HW/SW	5
Capitolul 3	5
Cerințe funcționale	5
Cerințe nefuncționale.....	7

Capitolul 1

Scopul proiectului

Proiectul propune dezvoltarea unei aplicații smart-home pentru controlul și gestionarea eficientă a dispozitivelor electronice din locuință.

Scopul acestei aplicații este de a oferi utilizatorilor un mod ușor și accesibil pentru a controla dispozitivele electronice din casă, precum iluminatul, temperatura, sistemele de securitate, electrocasnicele și multe altele. Prin intermediul acestei aplicații, utilizatorii pot controla și monitoriza dispozitivele din orice locație, prin intermediul aplicației.

Aplicația smart-home este concepută pentru a aduce confort și eficiență în viața de zi cu zi a utilizatorilor, oferind o soluție modernă și practică pentru gestionarea dispozitivelor electronice din casă.

Lista definițiilor

DCS – document cu cerințe software

Structura DCS

Documentul este împărțit în două capitole: capitolul 2 prezintă o descriere detaliată și o situație particulară, un exemplu al aplicației. Capitolul 3 prezintă cerințele funcționale/ nefuncționale ale produsului software.

Capitolul 2

Descrierea produsului software

Această aplicație de tip Smart Home este un produs software inovativ care permite utilizatorilor să controleze dispozitive inteligente din propria casă printr-o interfață de utilizator intuitivă și ușor de utilizat. Această aplicație este concepută să se adapteze nevoilor fiecărui utilizator în parte, oferindu-i posibilitatea de a adăuga, configura și controla diferite dispozitive din casă în mod individual.

Aplicația are o arhitectură client-server, unde serverul central este utilizat pentru a gestiona și monitoriza toate dispozitivele din rețea, în timp ce aplicația pentru utilizatorii finali este folosită pentru a interacționa cu serverul. Această arhitectură garantează securitatea și stabilitatea sistemului,

precum și posibilitatea de a adăuga dispozitive noi în rețea fără a fi necesară actualizarea aplicației.

Odată ce utilizatorul a creat un cont și s-a conectat la aplicație, acesta poate începe să adauge dispozitive inteligente la sistemul său Smart Home. Aceste dispozitive pot fi configurate și controlate prin intermediul aplicației, permitând utilizatorului să monitorizeze, să modifice și să îmbunătățească setările acestora în timp real.

Detalierea platformei HW/SW

Produsul software este dezvoltat pentru platforma sistemul de operare Windows 10 sau orice altă versiune, utilizând .NET Desktop Development pentru interfața grafică prezentată utilizatorului și mediul de dezvoltare Microsoft Visual Studio. Alte specificații ar include procesorul: 2.7 GHz sau mai rapid, Memoria 4GB RAM.

Capitolul 3

Cerințe funcționale

- 1) Logare / înregistrare client (utilizator sau admin):
 - Username și parolă unice la nivelul server-ului

- Posibilitatea reținerii datelor de autentificare
- Posibilitatea de log-out dacă utilizatorul dorește
- Cod special pentru crearea conturilor de tip admin

2) Client de tip USER:

- Poate să adauge/șteargă dispozitive
- Poate modifica parametrii de funcționare a dispozitivelor pe care le-a adăugat
- Alte funcționalități posibile: modificare parola/email , ștergere cont)

3) Client de tip ADMIN:

- Poate vizualiza toate dispozitivele din baza de date și le poate filtra după client sau tipul dispozitivului
- Poate modifica parametrii dispozitivelor
- Poate șterge conturi
- Poate vizualiza dispozitivele de tip log
- Promovarea conturilor de tip USER la conturi de tip ADMIN , sau le poate retrograda

4) Protocol:

- Informațiile care se transmit în cadrul aplicației se vor realiza pe suportul unui protocol propriu, în afară de cel implicit de rețea

- De exemplu, atunci când un client alege o anumită opțiune pusă la dispoziție (autentificare, înregistrare etc.) se va transmite o secvență specială de octeți pe care serverul va ști să o „înțeleagă” pentru a efectua operațiunile aferente .

Cerințe nefuncționale

- 1)Securitate: datele utilizatorilor și fișierele încărcate trebuie să fie securizate și protejate împotriva accesului neautorizat.
- 2)Performanță: aplicația trebuie să fie rapidă și să ofere o experiență fără probleme pentru utilizatori, chiar și atunci când numărul de utilizatori crește.
- 3)Interfața utilizator: interfața trebuie să fie intuitivă și ușor de utilizat pentru toți utilizatorii.
- 4)Scalabilitate: aplicația trebuie să poată fi scalată ușor pentru a gestiona creșterea numărului de utilizatori și a volumului de date.