**UNIVERSITATEA DE STAT DIN MOLDOVA**

**FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ**

**DEPARTAMENTUL DE INFORMATICĂ**

**GRĂDINARU MĂDĂLINA**

**DEZVOLTAREA UNUI WEBSITE AL UNEI COMPANII DE DESIGN INTERIOR**

**0613.4 INFORMATICĂ**

**Teză de an**

Șef de departament: Capcelea Titu, dr.,conf.univ

Conducător științific: Ulmanu Cristian

Autor: Grădinaru Mădălina

**CHIȘINĂU – 2024**

**MOLDOVA STATE UNIVERSITY**

**THE FACULTY OF MATHEMATICS AND COMPUTER SCIENCE**

**DEPARTMENT OF COMPUTER SCIENCE**

**GRĂDINARU MĂDĂLINA**

**DEVELOPMENT OF A WEBSITE OF AN INTERIOR DESIGN COMPANY**

**0613.4 Computer Science**

**Year thesis**

Head of the department : Capcelea Titu, dr.,conf.univ

Scientific leader: Ulmanu Cristian

Author: Grădinaru Mădălina

**CHIȘINĂU – 2024**

CUPRINS

[INTRODUCERE 4](#_Toc169989621)

[I. ANALIZA CERINȚELOR ȘI SPECIFICAȚIILOR 6](#_Toc169989622)

[1.1 Cerințe funcționale 6](#_Toc169989623)

[1.2 Cerințe non-funcționale 7](#_Toc169989624)

[II. PROIECTAREA SISTEMULUI 9](#_Toc169989625)

[2.1 Arhitectura sistemului 9](#_Toc169989626)

[2.2 Diagrama cazurilor de utilizare 9](#_Toc169989627)

[III. DEZVOLTAREA SITE-ULUI WEB PE PARTEA CLIENT 11](#_Toc169989628)

[3.1 HTML 11](#_Toc169989629)

[3.2 CSS & BOOTSTRAP5 11](#_Toc169989630)

[3.3 JavaScript 13](#_Toc169989631)

[V. DEZVOLTAREA SITE-ULUI WEB PE PARTEA SERVER 14](#_Toc169989632)

[5.1 Setarea mediului de dezvoltare 14](#_Toc169989633)

[5.1.1 OSPanel 14](#_Toc169989634)

[5.1.2 PHP 14](#_Toc169989635)

[5.1.3 Composer 15](#_Toc169989636)

[5.1.4 Laravel 15](#_Toc169989637)

[5.1.6 Baza de date 20](#_Toc169989638)

[5.1.7 Aimeos 21](#_Toc169989639)

[CONCLUZII 23](#_Toc169989640)

[REFERINȚE BIBLIOGRAFICE 24](#_Toc169989641)

# INTRODUCERE

În timpul actual al secolului XXI, prezența online a antreprenorilor a devenit nu doar esențială, ci necesară pentru a-și extinde impactul asupra societății deja digitalizate și pentru a atrage publicul larg asupra afacerii sale. Prin utilizarea celor mai recente tehnologii și tendinte de dezvoltare digitale, companiile se pot poziționa ca un lider inovator in lumea industrială, ceea ce îi oferă un spirit competitiv și relevant în acest sector aflat în continuă schimbare. Conform statisticilor recente, majoritatea consumatorilor cercetează spațiul online înainte de a lua decizii de cumpărare, inclusiv in sectorul designului interior. Anume acest comportament subliniaza importanța unei afișări bine structurate și profesionale către public.



*Sondaj bazat pe 30,000 de cumpărători la nivel mondial. Sursa: „What’s in a rewiew?”(Bazaarvoice.com)*

Așa cum designul interior este despre estetică, confort și calitate, aceste principii trebuie să se reflecte și în site-ul web a unei companii de profil. Astfel, prin intermediul acestui proiect, s-a urmărit dezvoltarea unui prototip de site web care să răspundă nevoilor specifice ale unei societăți comerciale de design interior ce poate contribui la consolidarea încrederii potențialilor clienți care își desfășoară căutările online și se bazează pe impresii digitale prin accesarea unor informații clare, navigare intuitivă și un design atrăgător.

Teza de față detaliază procesul de dezvoltare al site-ului cât și instrumentele de programare implementate. Obiectivele derivate din prezentul proiect includ:

* Crearea unei interfețe cât mai atractive, care să reflecte identitatea companiei și să ofere utilizatorilor o experiență plăcută,
* Integrarea unui portofoliu de proiecte de succes care să prezinte realizările companiei ,
* Asigurarea unei navigări ușoare și optimizate pentru utilizatorii mai mult sau mai puțin familiarizați cu tehnologiile moderne,
* Dezvoltarea paginii principale cât mai atractive, la fel și redirectionarea printr-un meniu către paginile secundare,
* Includerea posibilității de înregistrae/logare a clienților pentru posibilitatea de stocare securizată a datelor în scopul efectuării achizitiilor online din magazinul inclus, precum și pentru solicitatea anumitor servicii oferite de companie,
* Implementarea unui sistem de gestionare a resurselor care să permită administratorului să efectueze diferite manipulări cu produsele afișate utilizatorilor utilizînd o interfață grafică,
* Optimizarea paginii pentru a putea fi ușor detectată de motoarele de căutare, ceea ce ar însemna un număr mare de vizitatori.

Abordarea metodologică utilizată în teză este una mixtă, combinând cercetarea teoretică cu dezvoltarea practică. În principiu, se va baza pe o dezvoltare agilă, folosind următoarele instrumente:

* **HTML**- ca bază pentru structurarea coerentă a conținutului pe web,
* **CSS** - pentru controlul aspectului vizual a resurselor,
* **JavaScript** - pentru crearea de funcționalități interactive pentru experiența utilizatorilor,
* **PHP** - cu scopul de generare a conținutului dinamic, în cazul dat fiind **PostgreSQL**, utilizată pentru a asigura integritatea și securitatea datelor, dar și capacități de analiză și interogare avansate asupra datelor stocate.
* **PostgreSQL** - utilizată pentru a asigura integritatea și securitatea datelor, dar și capacități de analiză și interogare avansate asupra datelor stocate, folosind DBeaver pentru administrarea bazei de date,
* **Docker** - pentru crearea unui mediu de dezvoltare containerizat și portabil,
* **Laravel** - framework PHP utilizat pentru dezvoltarea rapidă și eficientă a aplicațiilor web,
* **Aimeos** - pentru integrarea unui sistem de e-commerce complet și flexibil în cadrul platformei Laravel.

Metodele de cercetare pe partea teoretică includ câteva aspecte, cum ar fi: analiza literaturii de specialitate privind designul web și optimizarea UX/UI, studiul comparativ al paginilor web de succes din acest domeniu, cercetarea și însușirea instrumentelor de dezvoltare web care ar permite utilizatorilor site-ului o experiență cât mai comodă și atrăgătoare de navigare.

Originalitatea proiectului curent constă în integrarea unor funcționalități precum grafice 3D și un aspect general luxos, demonstrând astfel modul în care tehnologiile moderne sunt un instrument important pentru a imbunătăți experiența utilizatorului și atingerea unor scopuri avansate de marketing online al companiei.

Valoarea aplicativă a rezultatelor obținute este eminentă în cazul companiilor ce au ca scop să-și îmbunătățească prezența online și să își extindă baza de clienți. Site-ul web curent din cadrul tezei va putea fi implementat și utilizat de către companiile de design interior reale pentru a-și promova activitatea, serviciile și produsele și, de asemenea, pentru a forma o interacțiune eficientă și profesională cu clienții.

Structura descrisă anterior, asigură o prezentare coerentă și comprehensivă a procesului complet a ceea ce în prezent se numește “web development” pentru o companie de design interior. Fiecare capitol ce urmează are menirea de a oferi o imagine clară asupra pașilor parcurși și a raționamentelor care au stat la baza proiectului prezent.

# I. ANALIZA CERINȚELOR ȘI SPECIFICAȚIILOR

## 1.1 Cerințe funcționale

În dezvoltarea site-ului web pentru o companie de design interior, cerințele funcționale sunt esențiale pentru a defini comportamentul și funcționalitățile necesare ale sistemului. Aceste cerințe au fost identificate pentru a asigura o experiență optimă pentru utilizatori și pentru a sprijini obiectivele companiei.

* Design atrăgător și informativ:

Pagina principală reprezintă prima impresie a companiei de design interior și reflectă identitatea sa vizuală distinctivă. Elementele principale includ un slider dinamic care prezintă imagini reprezentative ale proiectelor recente, oferind vizitatorilor o vedere rapidă și atractivă asupra diversității și calității serviciilor oferite. De asemenea, este inclusă o scurtă prezentare a echipei, evidențiind experiența și competențele membrilor companiei, ceea ce sporește încrederea și transparența în fața clienților potențiali. Structura paginii principale, precum și a tuturor celorlalte pagini, este intuitivă și ușor de navigat, facilitând accesul rapid la informațiile și serviciile esențiale ale companiei.

* Interacțiunea utilizatorilor cu site-ul:

Un aspect crucial al contactului este integrarea unui formular interactiv care permite paginilor să trimită recenzii și feedback direct companyi. Formula este concepută pentru a fi simplu și ușor de utilizat, facilitând comunicarea și interacțiunea directă cu utilizatorii site-ului. Utilizatorii pot informații necesare și pot trimite feedback-ul în mod eficient, asigurându-se că opinia lor este luată în considerare pentru completarea creării serviciilor oferite de companie. Această funcționalitate nu doar că sporește angajamentul utilizatorilor și implicarea acestora în experiența oferită de site-ul web, dar și contribuie la construirea unei relații de încredere și transparență între companii și clienți. Prin intermediul acestui formular, compania poate răspunde prompt la întrebările și preocupările lor, consolidând astfel încât să fie în piață și atrăgând noi oportunități de afaceri.

* Prezentarea portofoliului companiei:

Un punct forte reprezintă portofoliul companiei afișat într-o manieră vizuală atractivă, utilizând imagini de înaltă calitate și descrieri concise, dar detaliate, ale proiectelor. Astfel se pune accent pe estetica și inovațiile aduse în practică, oferind vizitatorilor o înțelegere clară a scopului fiecărui proiect și a soluțiilor implementate. De asemenea, prezentarea membrilor echipei și menționarea premiilor obținute contribuie la consolidarea autorității companiei în domeniul designului interior, evidențiind expertiza și succesul său în industrie.

* Funcționalități de e-commerce:

Funcționalități de e-commerce o componentă crucială în dezvoltarea site-ului pentru compania de design interior, asigurând o experiență de cumpărare fluidă și securizată pentru utilizatori. Integrarea unui magazin online intuitiv și ușor de navigat este esențială pentru vânzarea diverselor produse de design interior. Acesta va include categorii bine definite și opțiuni de filtrare avansată, facilitând găsirea rapidă a produselor dorite.

* Gestionarea coșului de cumpărături:

Acest compartiment al site-ului permite utilizatorilor să adauge și să gestioneze produsele în coșul de cumpărături într-un mod simplu și eficient. Utilizatorii pot vizualiza și modifica conținutul coșului înainte de a finaliza achiziția, asigurându-se astfel că toate detaliile sunt conforme cu preferințele lor.

* Checkout:

Procesul de checkout este conceput pentru a fi simplu și rapid, cu pași clari și intuitivi care ghidază utilizatorii în finalizarea comenzii. Metodele de plată disponibile sunt diverse și securizate, inclusiv opțiuni populare precum cardurile de credit, PayPal și altele, pentru a oferi flexibilitate maximă și încredere utilizatorilor în procesul de cumpărare online. Aceste funcționalități sunt integrate pentru a sprijini obiectivele comerciale ale companiei, contribuind la o experiență de utilizare plăcută și eficientă pe întreg parcursul procesului de cumpărare.

* Administrarea conținutului site-ului:

Panoul de administrare Aimeos permite administrarea completă a produselor de design interior disponibile în magazinul online. Administratorii pot adăuga noi produse cu imagini și descrieri detaliate, pot edita caracteristicile produselor existente pentru a reflecta schimbările de stoc sau preț, și pot șterge produsele care nu mai sunt disponibile sau care nu mai sunt în tendință.

## 1.2 Cerințe non-funcționale

În dezvoltarea site-ului web pentru compania de design interior !NDesign, este crucial să se ia în considerare cerințele non-funcționale pentru a asigura performanța, securitatea și o experiență generală de utilizare de înaltă calitate. Aceste cerințe sunt adesea la fel de importante ca cele funcționale și influențează modul în care site-ul este perceput și utilizat de către utilizatori.

Protejarea datelor utilizatorilor și a informațiilor companiilor este o prioritate deosebită. Site-ul trebuie să implementeze măsuri de securitate robustă pentru a preveni accesul neautorizat, furtul de date sau alte amenințări cibernetice. Acestea includ utilizarea conexiunilor securizate HTTPS, criptarea datelor sensibile și protecția împotriva vulnerabilităților cibernetice comune.

Design-ul responsiv este o caracteristică cheie, asigurând că site-ul se adaptează automat la diferite dimensiuni de ecrane, inclusiv pe dispozitive mobile, tablete și desktop-uri. Acest lucru contribuie la o experiență uniformă și plăcută pentru utilizatori, indiferent de dispozitivul pe care îl folosesc.

Compatibilitatea cu diverse browsere web a fost, de asemenea, un aspect crucial în dezvoltarea site-ului. Site-ul este optimizat pentru a funcționa corect pe cele mai recente versiuni ale principalelor browsere (Chrome, Firefox, Edge), asigurând o experiență consistentă și fără probleme de utilizare pentru toți vizitatorii.

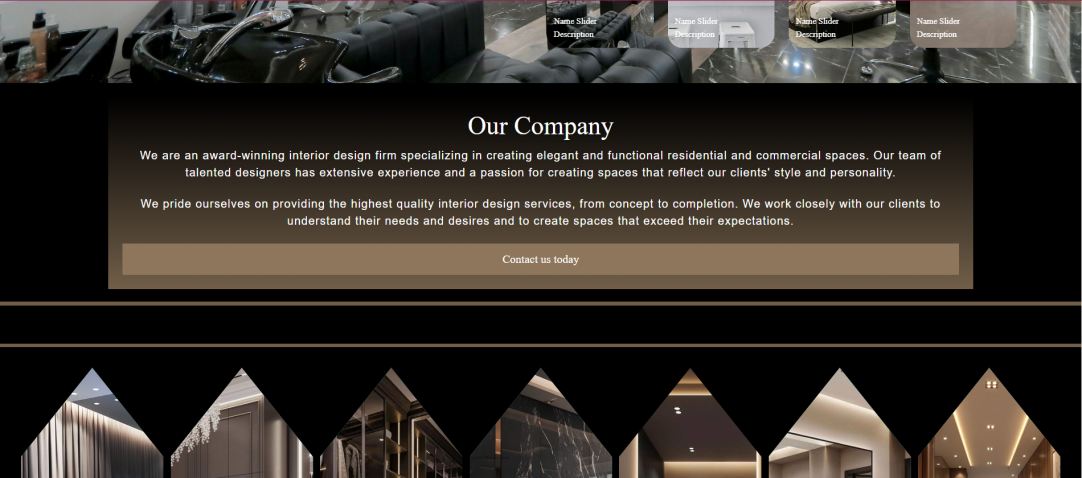
Interfața utilizatorului a fost proiectată să fie intuitivă și ușor de navigat. S-a pus accent pe un design economic și o structură clară a informațiilor, facilitând accesul rapid la conținutul relevant. În plus, alegerea culorilor și a fonturilor a fost făcută având în vedere normele de accesibilitate și practicile de UX/UI, pentru a asigura o experiență placută și eficientă în utilizare.

# II. PROIECTAREA SISTEMULUI

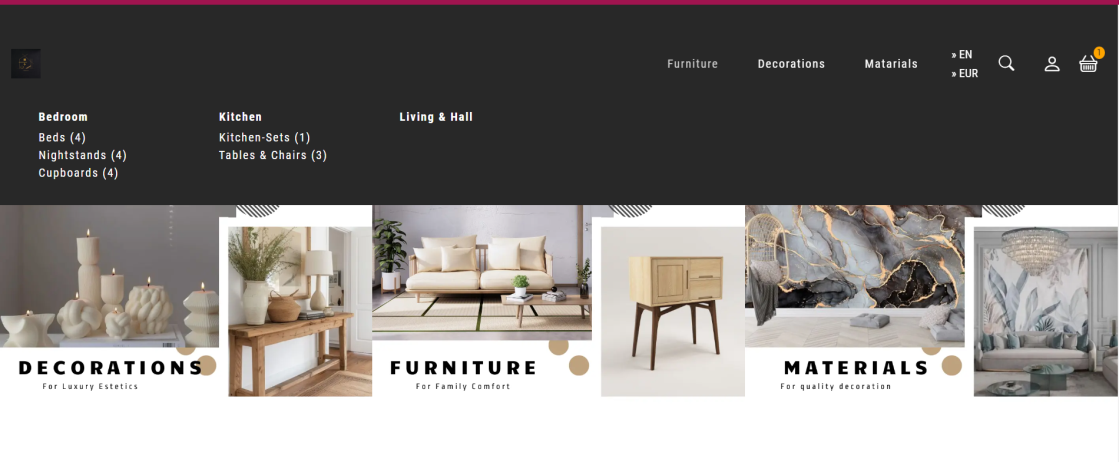
## 2.1 Arhitectura

La nivel de proiect, arhitectura poate fi logic împărțită în trei părți distincte pentru a asigura o funcționalitate eficientă și corespunzătoare:

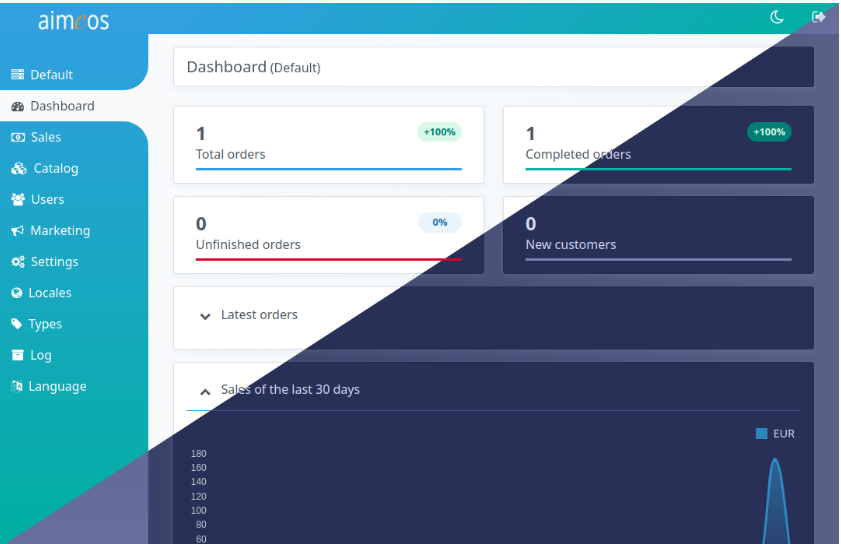
*Pagina de Prezentare a Companiei*: Această parte a site-ului web servește ca vitrină digitală pentru compania de design interior. Ea include informații despre identitatea vizuală a companiei, portofoliul său de proiecte de succes și echipa sa de experți. Scopul principal al acestei pagini este să atragă potențiali clienți și să stabilească o conexiune inițială solidă.



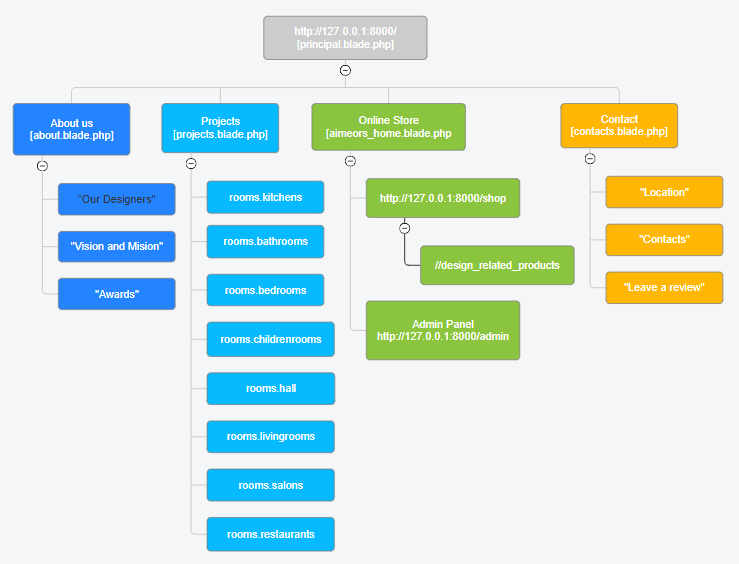
*Magazinul online:* Această parte permite utilizatorilor să exploreze și să achiziționeze produse de design interior direct de pe site-ul companiei. Aici sunt incluse funcționalități esențiale precum căutarea și filtrarea produselor, gestionarea coșului de cumpărături și un proces de checkout simplificat și securizat. Această secțiune optimizează experiența de cumpărare pentru clienți și sprijină obiectivele comerciale ale companiei.



*Pagina administratorului :* oferă funcționalități avansate pentru gestionarea eficientă a magazinului online. Administratorii pot adăuga, edita și șterge produse, gestiona stocurile și prețurile, monitoriza comenzile și interacționa cu clienții. Această parte a sistemului asigură controlul complet și eficient al operațiunilor interne ale companiei.



Arhitectura sistemului reprezintă coloana vertebrală a oricărei aplicații software, definind structura și organizarea componentelor sale pentru a asigura funcționalitatea, performanța și securitatea dorită. În contextul dezvoltării unui site web pentru o companie de design interior, arhitectura sistemului joacă un rol crucial în crearea unei platforme robuste și eficiente, capabilă să îndeplinească cerințele complexe ale utilizatorilor și ale afacerii.

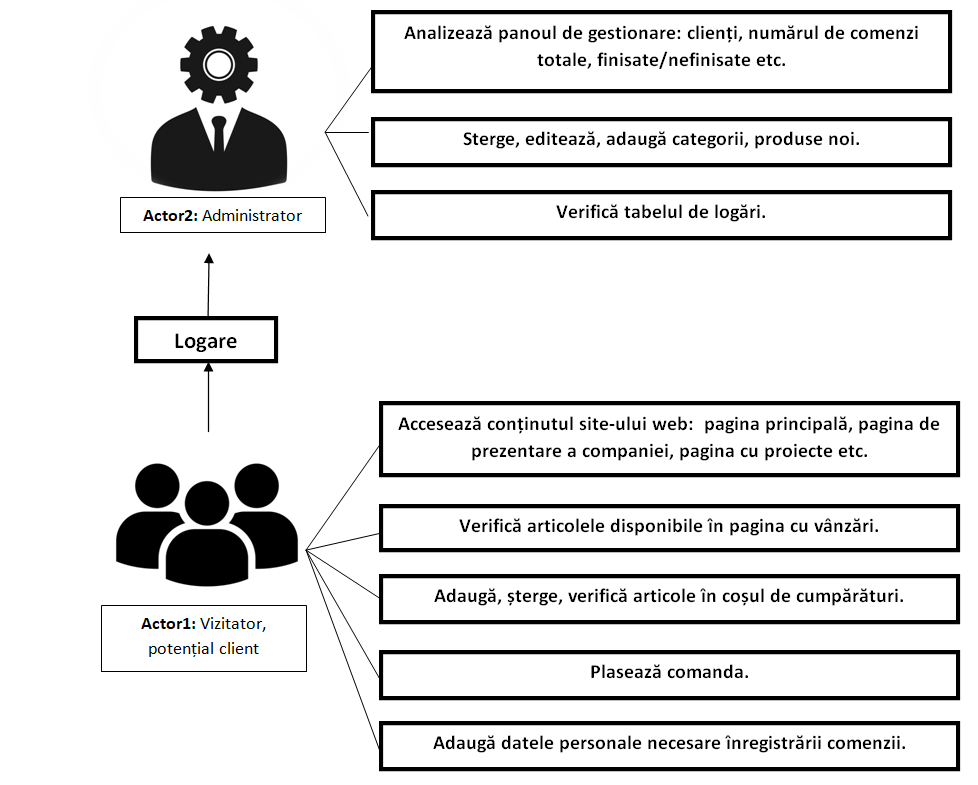


## 2.2 Diagrama cazurilor de utilizare

O diagramă a cazurilor de utilizare (Use Case Diagram - UCD) este un instrument fundamental pentru planificarea și proiectarea unui site web de design interior eficient. Această diagramă oferă o reprezentare vizuală a modului în care diferiți utilizatori vor interacționa cu site-ul web pentru a-și îndeplini obiectivele.

Diagrama cazurilor de utilizare ilustrează interacțiunile dintre doi actori principali - vizitatori și administratori - și un site web. Vizitatorii pot naviga prin conținut, pot vizualiza produse, pot gestiona coșul de cumpărături și pot finaliza comenzi. Administratorii pot gestiona site-ul prin accesarea panoului de administrare, adăugarea/editarea/ștergerea de produse și vizualizarea jurnalelor. Diagrama oferă o imagine de ansamblu a funcționalităților de bază și servește ca punct de plecare pentru o analiză detaliată a cerințelor și a designului sistemului.

Fig.



Această diagramă oferă o imagine clară a funcționalităților site-ului web, facilitează comunicarea și colaborarea în cadrul echipei de dezvoltare și permite identificarea și remedierea potențialelor probleme încă din fazele incipiente ale proiectului.

# III. DEZVOLTAREA SITE-ULUI WEB PE PARTEA CLIENT

Front-end-ul, elementul fundamental al interacțiunii digitale, constituie scena principală a unui site web sau a unei aplicații web, cu care utilizatorii se conectează direct. Depășind rolul de simplă prezentare statică, front-end-ul modern oferă o experiență interactivă și dinamică, facilitând navigarea, interacțiunea și îndeplinirea obiectivelor utilizatorilor.

Proiectul dat demonstrează o abordare atentă a dezvoltării front-end, utilizând o combinație de tehnologii moderne și strategii eficiente pentru a crea o interfață atractivă și funcțională

## 3.1 HTML

Limbajul fundamental al internetului, HTML permite definirea structurii și a conținutului paginilor web, creând o bază solidă pentru interfața utilizatorului. Documentele HTML sunt compuse din blocuri de construcție numite elemente, fiecare reprezentat de etichete închise în paranteze unghiulare (< >). Aceste elemente definesc diferitele componente ale unei pagini web, cum ar fi titluri, paragrafe, liste, imagini și linkuri. Prin combinarea acestor elemente, dezvoltatorii construiesc structura ierarhică a unei pagini web.

Codul HTML furnizat definește o secțiune pe o pagină web dedicată prezentării descrierii companiei. Secțiunea include un subtitlu, două paragrafe de text și un hyperlink către pagina de contact.

 <div class="company-description">

  <h5>Our Company</h5>

  <p>We are an award-winning interior design firm specializing in creating elegant and functional residential and commercial spaces. Our team of talented designers has extensive experience and a passion for creating spaces that reflect our clients' style and personality.</p>

  <p>We pride ourselves on providing the highest quality interior design services, from concept to completion. We work closely with our clients to understand their needs and desires and to create spaces that exceed their expectations.</p>

  <a href="/contact/">Contact us today</a>

</div>

## 3.2 CSS & BOOTSTRAP

Cascading Style Sheets (CSS) joacă un rol crucial în proiectul meu, oferind control granular asupra aspectului și prezentării paginilor web. Prin intermediul CSS, pot defini fonturi, culori, spațiere, layout și alte elemente vizuale care contribuie la crearea unei experiențe de utilizator coerente și estetice.

Utilizarea CSS permite separarea conținutul de prezentare, facilitând menținerea și actualizarea codului. Modificările aduse fișierelor CSS pot afecta instantaneu aspectul întregului site web, fără a fi necesară modificarea codului HTML individual pentru fiecare element.

De asemenea, CSS oferă flexibilitatea de a adapta aspectul site-ului web la diferite dispozitive și dimensiuni ale ecranului. Prin intermediul media queries, se pot defini reguli specifice pentru ecrane mobile, tablete și desktop-uri, asigurând o experiență optimă pentru toți utilizatorii.

@media screen and (max-width: 678px) {

    .carousel .list .item .content{

        padding-right: 0;

    }

    .carousel .list .item .content .title{

        font-size: 30px;

    }

}

Utilizarea Bootstrap, framework CSS, în dezvoltarea front-end este esențială pentru crearea de site-uri web moderne și responsive. În exemplul de cod prezentat, Bootstrap asigură adaptarea automată a elementelor la diferite dimensiuni ale ecranului prin clasa container-fluid, facilitând astfel o experiență consistentă pe toate dispozitivele.

<nav class="navbar navbar-expand-lg navbar-dark custom-navbar container-fluid">

    <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">

        <span class="navbar-toggler-icon"></span>

    </button>

    <div class="collapse navbar-collapse na" id="navbarNav">

        <ul class="navbar-nav">

        <a class="navbar-brand"  href="{{ route('principal') }}">!NDesign</a>

        <li class="nav-item active">

                <a class="nav-link" href="{{ route('principal') }}">Home</a>

            </li>

            <li class="nav-item">

            <a class="nav-link" href="{{ url('about') }}">About Us</a>

            </li>

            <li class="nav-item">

                <a class="nav-link" href="{{ url('projects') }}">Projects</a>

            </li>

            <li class="nav-item ">

                <a class="nav-link" href="{{ route('aimeos\_home') }}">Online Store</a>

            </li>

            <li class="nav-item">

                <a class="nav-link" href="{{ url('contacts') }}">Contact</a>

            </li>

        </ul>

Componentele predefinite, precum navbar, navbar-toggler, collapse și navbar-nav, simplifică construirea unor bare de navigare complexe și responsive. De asemenea, utilizarea claselor standardizate garantează un design uniform și ușor de întreținut. Bootstrap permite dezvoltarea rapidă și eficientă a interfețelor de utilizator, economisind timp și reducând efortul necesar pentru implementarea designurilor responsive și interactive. Compatibilitatea framework-ului cu majoritatea browserelor moderne asigură o experiență fără probleme pentru utilizatori. Astfel, Bootstrap accelerează dezvoltarea și permite dezvoltatorilor să se concentreze pe funcționalitatea specifică a aplicației.

## 3.3 JavaScript

JavaScript joacă un rol crucial în proiect, oferind funcționalitate interactivă și dinamism.

* Permite utilizatorilor să interacționeze cu elementele paginii, cum ar fi butoane, formulare, meniuri și galerii de imagini.
* Creează o experiență mai captivantă și mai receptivă, sporind implicarea utilizatorilor.
* Permite implementarea de acțiuni dinamice, cum ar fi clicuri, hover-uri și trageri.

Codul JavaScript din exemplu demonstrează modul în care JavaScript este utilizat pentru a controla un carusel de imagini. Funcțiile nextDom.onclick și prevDom.onclick sunt declanșate atunci când utilizatorul face clic pe butoanele "următor" și "anterior", respectiv. Aceste funcții utilizează funcția showSlider pentru a afișa următoarea sau precedenta imagine din carusel.

nextDom.onclick = function(){

    showSlider('next');

}

prevDom.onclick = function(){

    showSlider('prev');

}

# V. DEZVOLTAREA SITE-ULUI WEB PE PARTEA SERVER

Back-end-ul, nucleul funcțional al interacțiunii digitale, este elementul cheie care alimentează și susține funcționalitatea unui site web sau a unei aplicații web. Mai mult decât un simplu mecanism de stocare și prelucrare a datelor, back-end-ul modern oferă infrastructura necesară pentru gestionarea eficientă a proceselor complexe și asigurarea unei experiențe fluide pentru utilizatori. În acest proiect, se demonstrează o abordare meticuloasă a dezvoltării back-end, utilizând o combinație de tehnologii avansate și practici eficiente pentru a crea un sistem robust și performant, capabil să răspundă nevoilor variate și dinamice ale utilizatorilor.

În dezvoltarea aplicațiilor moderne, precum și a prezentului website, utilizarea unui set diversificat de tehnologii și instrumente este crucială pentru a asigura performanță, scalabilitate și securitate. În acest context, o combinație bine echilibrată de framework-uri, baze de date și instrumente de dezvoltare devine fundamentala.

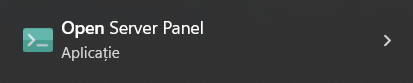
## 

## 5.1 Setarea mediului de dezvoltare

Pentru a asigura dezvoltarea eficientă și robustă a aplicației !NDesign, este esențială configurarea unui mediu de dezvoltare bine structurat. Aceasta implică utilizarea unei suite de instrumente și tehnologii care facilitează gestionarea dependențelor, rularea serverelor web, gestionarea bazelor de date și dezvoltarea codului într-un mod colaborativ și controlat.

## 5.1.1 OSPanel

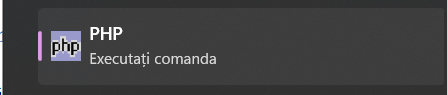
OSPanel este un pachet de server local care permite rularea unui server web complet pe mașina locală. Acesta include diverse servicii precum Apache, MySQL și PHP, permițând dezvoltatorilor să testeze și să dezvolte aplicații web în mod local. Instalarea și configurarea OSPanel sunt simple și rapide, făcându-l ideal pentru dezvoltatorii care doresc să creeze un mediu de testare eficient.



open\_server\_panel\_6\_0\_0\_setup.exe

## 5.1.2 PHP

PHP (Hypertext Preprocessor) este un limbaj de programare open-source, popular pentru dezvoltarea aplicațiilor web dinamice și interactive. Este unul dintre cele mai utilizate limbaje de scripting de pe web, fiind integrat frecvent în serverele web pentru generarea de pagini web dinamice.



PHP este esențial în dezvoltarea backend-ului pentru aplicații web moderne, oferind un echilibru între putere, flexibilitate și ușurință în utilizare. Utilizarea sa în combinație cu framework-uri PHP precum Laravel și platforme de ecommerce precum Aimeos, contribuie la crearea unor aplicații web robuste și performante.

## 5.1.3 Composer

Un pas important de pornire în dezvoltarea !NDesign l-a constituit Composer, un instrument puternic de gestionare a dependențelor în mediul PHP. Prin intermediul fișierului composer.json este posibilă definirea și instalarea automată a bibliotecilor și pachetelor necesare proiectului.

php composer-setup.php --install-dir=bin --filename=composer



Acest lucru a simplificat semnificativ procesul de integrare a componentelor externe și de menținere a coerenței versiunilor, facilitând astfel dezvoltarea și îmbunătățirea continuă a aplicației. Utilizând Composer, a fost asigurat că toate componentele folosite în !NDesign sunt actualizate și compatibile, contribuind astfel la stabilitatea și performanța aplicației în timp.

## 5.1.4 Laravel

În timp ce Composer asigură gestionarea eficientă a dependențelor, Laravel utilizează aceste pachete pentru a oferi o infrastructură robustă și unelte puternice pentru dezvoltarea rapidă a aplicațiilor web moderne.

Astfel, Laravel reprezintă un framework PHP open-source, cunoscut pentru simplitatea și eleganța sa, construit cu scopul de a facilita dezvoltarea aplicațiilor web moderne și performante. Prin abordarea sa orientată pe dezvoltator, Laravel pune accent pe productivitate și claritatea codului, oferind un set bogat de funcționalități esențiale pentru dezvoltarea rapidă și eficientă a aplicațiilor web.

Framework-ul Laravel se bazează pe principiul "*convenției în locul configurației"*, ceea ce înseamnă că oferă structuri predefinite și convenționale pentru sarcini comune în dezvoltarea web, cum ar fi rutarea, gestionarea bazelor de date, autentificarea utilizatorilor și multe altele. Aceasta face ca dezvoltarea să fie mai rapidă și mai intuitivă, reducând timpul necesar pentru configurarea și gestionarea detaliilor tehnice.

Fișierul **.env** în Laravel este un element central al configurării aplicației, utilizat pentru gestionarea variabilelor de mediu specifice fiecărei instanțe de aplicație. Acesta conține setările necesare pentru diferite aspecte ale aplicației, cum ar fi configurările bazei de date, setările de sesiune, setările de autentificare SMTP și alte variabile de configurare specifice aplicației. Iată o descriere detaliată a variabilelor din exemplul dat:

* **APP\_NAME=Aimeos**: Numele aplicației.
* **APP\_ENV=local**: Aplicația rulează într-un mediu local de dezvoltare.
* **APP\_KEY=base64+KSCb799g=**: Cheia de aplicație este o valoare encodată în base64, utilizată pentru criptarea datelor sensibile.
* **APP\_DEBUG=1**: Modul de depanare este activat în dezvoltare (1 pentru activat, 0 pentru dezactivat).
* **APP\_VERSION=1**: Versiunea aplicației este setată la "1".
* **APP\_URL=**[**http://localhost:8000**](http://localhost:8000): URL-ul de bază al aplicației.
* **SHOP\_MULTILOCALE=**: Setare pentru multi-localizare în cadrul magazinului, nu este specificată o valoare.
* **SHOP\_MULTISHOP=**: Setare pentru gestionarea mai multor magazine, nu este specificată o valoare.
* **SHOP\_REGISTRATION=**: Setare pentru înregistrarea utilizatorilor în magazin, nu este specificată o valoare.
* **SHOP\_PERMISSION=admin**: Permisiuni de administrare în cadrul magazinului sunt setate pentru rolul "admin".
* **LOG\_CHANNEL=stack**: Canalele de logare sunt configurate pentru a utiliza stiva de canale.
* **DB\_CONNECTION=pgsql**: Conexiunea la baza de date este configurată pentru PostgreSQL.
* **DB\_HOST=127.0.0.1**: Adresa IP a gazdei pentru serverul de bază de date este setată la 127.0.0.1 (localhost).
* **DB\_PORT=5432**: Portul pentru conexiunea la serverul PostgreSQL este setat la 5432.
* **DB\_DATABASE=design**: Numele bazei de date utilizată de aplicație este "design".
* **DB\_USERNAME=postgres**: Utilizatorul pentru conectarea la baza de date PostgreSQL este "postgres".
* **DB\_PASSWORD=mysecretpassword**: Parola utilizatorului pentru conectarea la baza de date este "mysecretpassword".
* **BROADCAST\_DRIVER=log**: Driver-ul pentru transmiterea evenimentelor este configurat pentru înregistrarea în loguri.
* **CACHE\_DRIVER=file**: Driver-ul de cache este configurat pentru a utiliza sistemul de fișiere.
* **QUEUE\_CONNECTION=sync**: Conexiunea la cozi de procesare este setată pentru procesare sincronă.
* **SESSION\_DRIVER=file**: Driver-ul pentru gestionarea sesiunilor utilizatorilor este configurat pentru a utiliza sistemul de fișiere.
* **SESSION\_LIFETIME=120**: Durata de viață a sesiunilor utilizatorilor este setată la 120 de minute.
* **REDIS\_HOST=127.0.0.1**: Adresa IP a gazdei pentru serverul Redis este setată la 127.0.0.1 (localhost).
* **REDIS\_PASSWORD=**: Parola pentru serverul Redis nu este specificată.
* **REDIS\_PORT=6379**: Portul pentru conexiunea la serverul Redis este setat la 6379.
* **MAIL\_MAILER=smtp**: Sistemul de trimitere a emailurilor este configurat pentru a utiliza SMTP.
* **MAIL\_HOST=smtp.mailtrap.io**: Adresa serverului SMTP pentru trimiterea emailurilor este setată la smtp.mailtrap.io.
* **MAIL\_PORT=2525**: Portul serverului SMTP este setat la 2525.
* **MAIL\_USERNAME=**: Utilizatorul pentru conexiunea la serverul SMTP nu este specificat.
* **MAIL\_PASSWORD=vzdchzrynicpetsg**: Parola pentru conexiunea la serverul SMTP este setată la "vzdchzrynicpetsg".
* **MAIL\_ENCRYPTION=tls**: Conexiunea la serverul SMTP este criptată utilizând TLS.
* **MAIL\_FROM\_ADDRESS=**: Adresa de email pentru expedierea emailurilor nu este specificată.
* **AWS\_ACCESS\_KEY\_ID=**, **AWS\_SECRET\_ACCESS\_KEY=**, **AWS\_DEFAULT\_REGION=**, **AWS\_BUCKET=**: Configurări pentru integrarea cu serviciile AWS nu sunt specificate.
* **PUSHER\_APP\_ID=**, **PUSHER\_APP\_KEY=**, **PUSHER\_APP\_SECRET=**, **PUSHER\_APP\_CLUSTER=**: Configurări pentru integrarea cu Pusher nu sunt specificate.
* **MIX\_PUSHER\_APP\_KEY=**, **MIX\_PUSHER\_APP\_CLUSTER=**: Configurări pentru front-end generate automat în timpul instalării pentru a configura comunicarea cu Pusher.

Laravel utilizează arhitectura Model-View-Controller (MVC), care promovează o separare clară a preocupărilor și un cod mai modular. MVC separă logica de afaceri (modele) de interfața utilizatorului (vizualizări) și de controlul fluxului de aplicație (controlere).

*Modelele* reprezintă logica de afaceri și interacțiunile cu baza de date. În Laravel, modelele sunt utilizate pentru a interacționa cu tabelele din baza de date, permițând dezvoltatorilor să execute interogări și să gestioneze datele într-un mod orientat pe obiecte.

Exemplu de model utilizat în aplicație:

* Clasa User extinde Authenticatable, ceea ce o face capabilă să utilizeze funcționalitățile de autentificare oferite de Laravel.

class User extends Authenticatable implements MustVerifyEmail

{

    use HasApiTokens, HasFactory, Notifiable;

* Atributul $fillable definește câmpurile care pot fi completate în mod masiv. Acest lucru ajută la prevenirea atacurilor de asignare în masă, permițând doar anumite câmpuri să fie completate prin intermediul metodelor create sau update.

/\*\*

     \* The attributes that are mass assignable.

     \*

     \* @var array<int, string>

     \*/

    protected $fillable = [

        'name',

        'email',

        'password',

        'siteid',

    ];

*Vizualizările* sunt responsabile pentru prezentarea interfeței utilizatorului. În Laravel, vizualizările sunt fișiere Blade (.blade.php) care permit generarea dinamică a conținutului HTML, folosind șabloane și datele transmise de controlere.

* Blade, motorul de template-uri al Laravel, permite crearea și reutilizarea componentelor vizuale, reducând duplicarea codului și facilitând modificările ulterioare. De exemplu, o componentă Blade pentru un formular de autentificare poate fi reutilizată în mai multe locuri din aplicație:

*Controlerele* gestionează fluxul aplicației și coordonează interacțiunea dintre modelele și vizualizările aplicației. În Laravel, controlerele sunt responsabile pentru gestionarea cererilor HTTP, executarea logicii de afaceri și returnarea răspunsurilor corespunzătoare.

În acest exemplu, ReviewController din Laravel servește ca un punct central pentru gestionarea operațiilor asociate evaluărilor (reviews) utilizatorilor.

<?php

namespace App\Http\Controllers;

use Illuminate\Http\Request;

use App\Models\Review;

class ReviewController extends Controller

{

    public function store(Request $request)

    {

        $request->validate([

            'name' => 'required|string|max:255',

            'email' => 'required|string|email|max:255',

            'rating' => 'required|integer|min:1|max:5',

            'comment' => 'required|string',

        ]);

        Review::create([

            'name' => $request->name,

            'email' => $request->email,

            'rating' => $request->rating,

            'comment' => $request->comment,

        ]);

        return redirect()->route('contacts')->with('success', 'Review submitted successfully.');

    }

}

ReviewController în Laravel exemplifică utilizarea controlerelor pentru gestionarea acțiunilor asociate evaluărilor utilizatorilor într-o aplicație web. În metoda store, se validează și se asigură integritatea datelor primite prin cererea HTTP (Request). Dacă datele sunt validate cu succes conform regulilor definite, o nouă înregistrare este creată în baza de date folosind modelul Review. Acest proces include salvarea numelui, emailului, evaluării și comentariului utilizatorului. La finalizarea procesului, utilizatorul este redirecționat către ruta contacts cu un mesaj de succes, notificându-l că evaluarea a fost trimisă cu succes. Această abordare demonstrează modul în care controlerele facilitează gestionarea datelor și comunicarea între utilizatori și aplicație în Laravel.

În Laravel, *sistemul de rute* constituie un pilon fundamental al arhitecturii aplicațiilor web. Aceste rute definesc legătura între URL-uri și codul aplicației care le gestionează, stabilind fluxul și comportamentul aplicației în funcție de cererile utilizatorilor. Introducerea lor este esențială pentru orice aplicație Laravel, oferind un mod clar și organizat de a dirija solicitările HTTP către acțiuni specifice. Fiecare rută poate să afișeze conținut din vederi, să apeleze metode din controlere sau să gestioneze alte logici specifice aplicației. Prin intermediul sistemului de rutare Laravel, dezvoltatorii pot crea aplicații web puternice și flexibile, beneficiind de o structură clară și modulară pentru gestionarea fluxului de date și interacțiunilor utilizatorilor.

Route::group(['middleware' => ['web']], function () {

    Route::get('/shop', [CatalogController::class, 'homeAction'])->name('aimeos\_home');

});

În exemplul dat, este definită o rută pentru URL-ul /shop, care utilizează metoda homeAction din controller-ul CatalogController pentru a gestiona cererea. Ruta este numită 'aimeos\_home' și utilizează middleware-ul 'web'.

*Middleware*-urile în Laravel sunt straturi intermediare de logică utilizate pentru a gestiona și filtra cererile HTTP înainte sau după ce acestea ajung la aplicația principală. Acestea sunt esențiale pentru implementarea autentificării, autorizării, validării datelor de intrare și gestionării sesiunilor utilizatorilor. De asemenea, pot adăuga antete HTTP pentru securitate, loga informații pentru auditare și optimiza răspunsurile prin compresie sau cache.

 protected $middlewareGroups = [

        'web' => [

            \App\Http\Middleware\EncryptCookies::class,

            \Illuminate\Cookie\Middleware\AddQueuedCookiesToResponse::class,

            \Illuminate\Session\Middleware\StartSession::class,

            \Illuminate\View\Middleware\ShareErrorsFromSession::class,

            \App\Http\Middleware\VerifyCsrfToken::class,

            \Illuminate\Routing\Middleware\SubstituteBindings::class,

        ],

 protected $middlewareAliases = [

        'auth' => \App\Http\Middleware\Authenticate::class,

        'auth.basic' => \Illuminate\Auth\Middleware\AuthenticateWithBasicAuth::class,

        'auth.session' => \Illuminate\Session\Middleware\AuthenticateSession::class,

        'cache.headers' => \Illuminate\Http\Middleware\SetCacheHeaders::class,

        'can' => \Illuminate\Auth\Middleware\Authorize::class,

        'guest' => \App\Http\Middleware\RedirectIfAuthenticated::class,

        'password.confirm' => \Illuminate\Auth\Middleware\RequirePassword::class,

        'signed' => \Illuminate\Routing\Middleware\ValidateSignature::class,

        'throttle' => \Illuminate\Routing\Middleware\ThrottleRequests::class,

        'verified' => \App\Http\Middleware\EnsureEmailIsVerified::class,

        'setlocale' => \App\Http\Middleware\SetLocale::class,

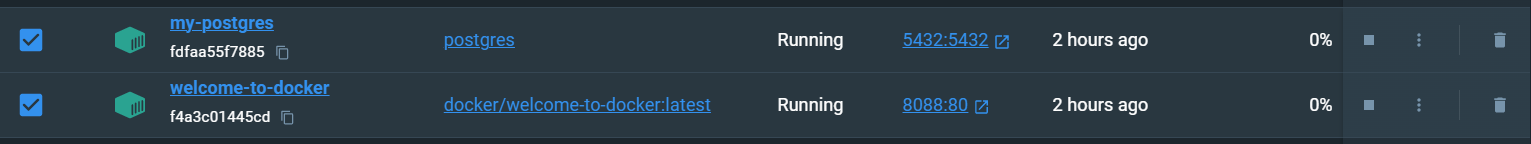
    ];

În fișierul Kernel.php din Laravel, middleware-urile joacă un rol crucial în gestionarea cererilor HTTP în aplicație. Middleware-urile globale, definite în $middleware, sunt aplicate în fiecare cerere și includ diverse funcționalități cum ar fi gestionarea CORS, verificarea dimensiunii datelor POST și conversia șirurilor goale în valori nule. În grupurile de middleware, cum ar fi 'web' pentru rutele web și 'api' pentru rutele API, sunt incluse procese specifice precum gestionarea sesiunilor, verificarea token-urilor CSRF, și înlocuirea legăturilor. Middleware-urile pot fi de asemenea atribuite direct rutelor sau grupurilor de rute pentru a aplica logica specifică în funcție de necesități, cum ar fi autentificarea, autorizarea, cache-ul și verificarea statutului de autentificare a sesiunii utilizatorului. Aceste configurări în Kernel.php permit Laravel să ofere o gestionare flexibilă și eficientă a cererilor HTTP în aplicații web moderne.

## 5.1.6 Baza de date

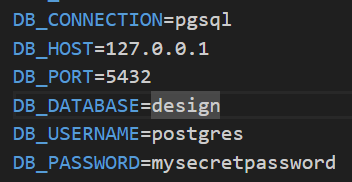
În cadrul proiectului, baza de date reprezintă un pilon esențial pentru stocarea și gestionarea datelor critice, inclusiv informațiile despre utilizatori, produse, comenzi și alte entități relevante pentru funcționarea aplicației.

Utilizarea *Docker* în cadrul proiectului a fost crucială pentru administrarea și gestionarea eficientă a PostgreSQL. Docker oferă un mediu izolat pentru rularea PostgreSQL în containere, ceea ce permite dezvoltatorilor să creeze și să testeze aplicația într-un mod consistent, indiferent de configurațiile locale. Prin Docker, se asigură că toate dependențele necesare, inclusiv PostgreSQL, sunt instalate și configurate într-un mod uniform și ușor de gestionat.

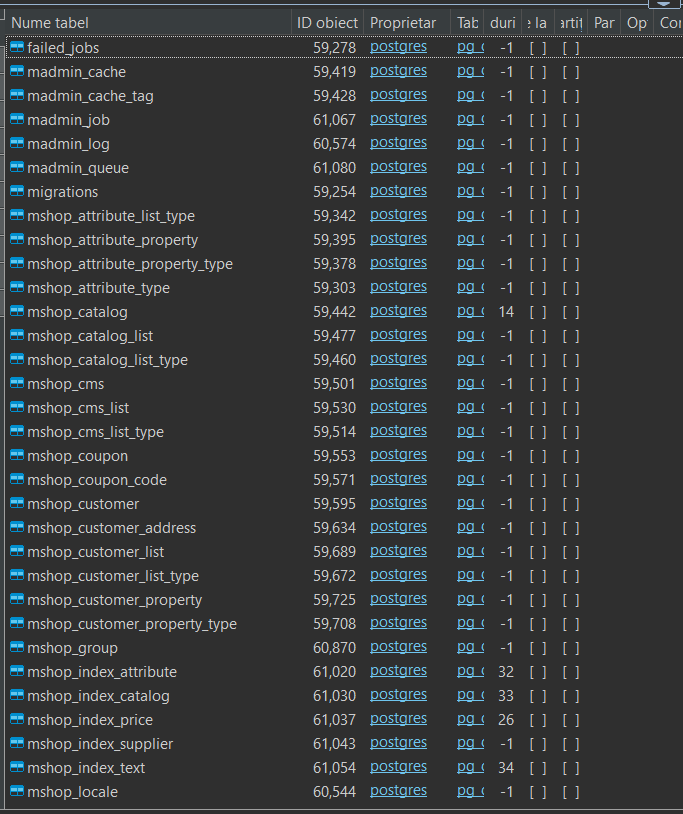


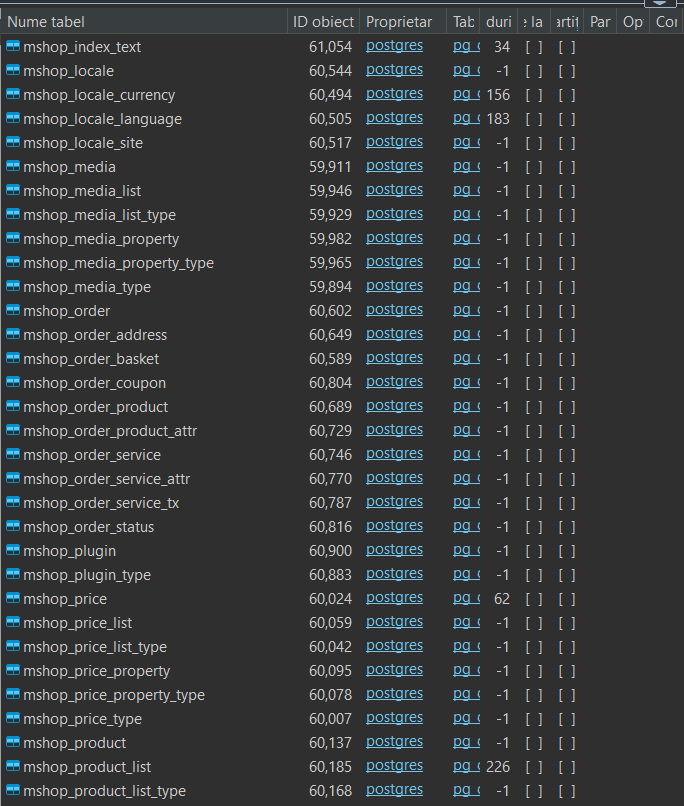
Pe de altă parte, *DBeaver* a fost alegerea ideală pentru administrarea și interacțiunea eficientă cu PostgreSQL. DBeaver oferă o interfață intuitivă și puternică pentru gestionarea schemelor bazelor de date, executarea interogărilor complexe și vizualizarea datelor. Acest instrument permite echipei să exploreze și să modifice structura bazei de date într-un mod flexibil și eficient, contribuind la optimizarea performanței și fiabilității aplicației.

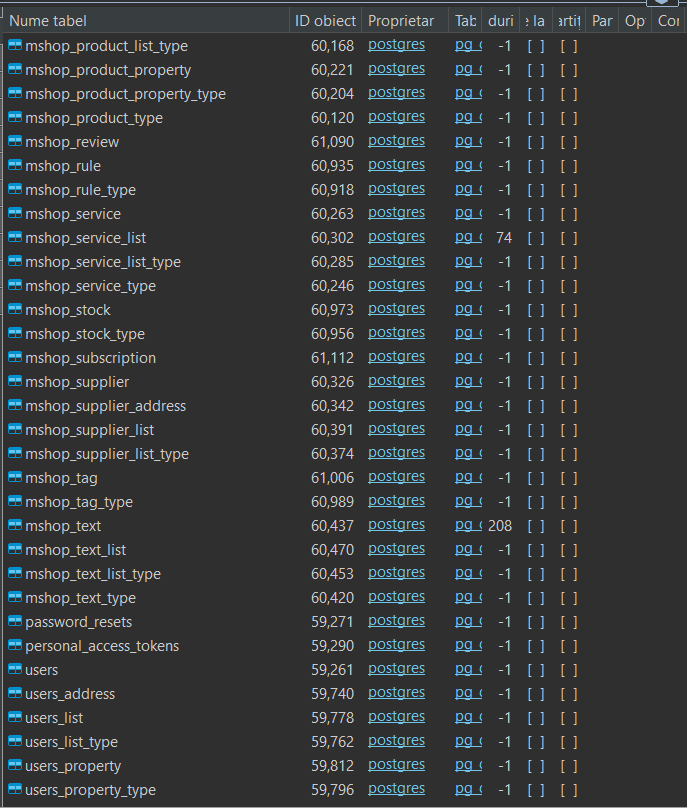
Configurația bazei de date proiectul Laravel a fost realizat prin setările din fișierul .env:



Aceste setări permit Laravel să stabilească o conexiune eficientă și sigură cu baza de date *PostgreSQL*, esențială pentru a efectua operațiuni de citire și scriere în cadrul aplicației web. Prin utilizarea directă a acestor setări în fișierul .env, Laravel automatizează gestionarea conexiunii și interacțiunii cu baza de date, oferind dezvoltatorilor un mediu de lucru consistent și ușor de configurat.







## 5.1.7 Aimeos

Aimeos este un framework de e-commerce de înaltă performanță, cunoscut pentru flexibilitatea sa remarcabilă și capacitatea de scalare. Această platformă oferă soluții avansate pentru dezvoltarea de magazine online, fiind capabilă să gestioneze volume mari de date și traficul asociat acestora. Caracteristicile sale arhitecturale și posibilitățile de personalizare îl fac ideal pentru proiecte de e-commerce complexe și personalizate.[]

Integrarea Aimeos în Laravel este un proces direct datorită pachetelor disponibile prin Composer. Această integrare oferă dezvoltatorilor toate avantajele Laravel, cum ar fi routing-ul ușor de configurat, sistemul de migrații și motorul de șabloane Blade, combinat cu puterea și flexibilitatea Aimeos.

În procesul de instalare a Aimeos într-un proiect Laravel, configurarea corectă a bazei de date și migrarea tabelelor necesare sunt pași cruciali. Aceștia asigură că structura bazei de date este pregătită pentru a gestiona datele magazinului online, cum ar fi produsele, comenzile și utilizatorii. Procesul complet de pregătire a bazei de date implică următorii pași:

1. Configurare .env:
2. DB\_CONNECTION=pgsql
3. DB\_HOST=127.0.0.1
4. DB\_PORT=5432
5. DB\_DATABASE=design
6. DB\_USERNAME=postgres
7. DB\_PASSWORD=mysecretpassword
8. Rularea Migrațiilor Laravel:

*C:\company>php artisan migrate*

1. Rularea Migrațiilor Aimeos:

*C:\company>php artisan aimeos:setup*

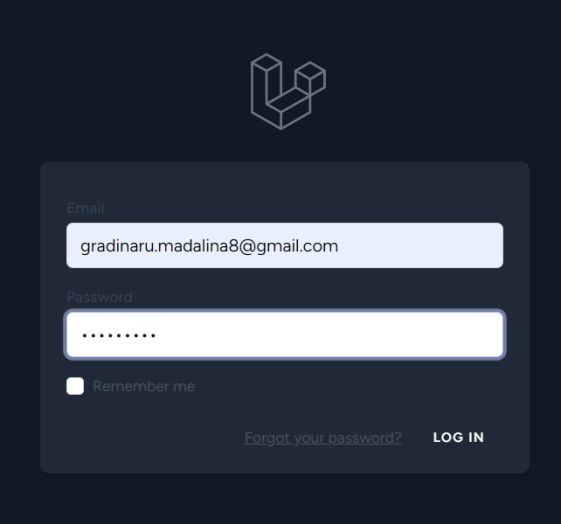
Prin urmarea acestor pași, se asigură că baza de date este pregătită pentru a gestiona datele magazinului online, că toate tabelele necesare sunt create corect și că structura bazei de date este compatibilă cu schema utilizată de Aimeos și Laravel. Aceasta garantează o funcționare optimă și sigură a magazinului online.

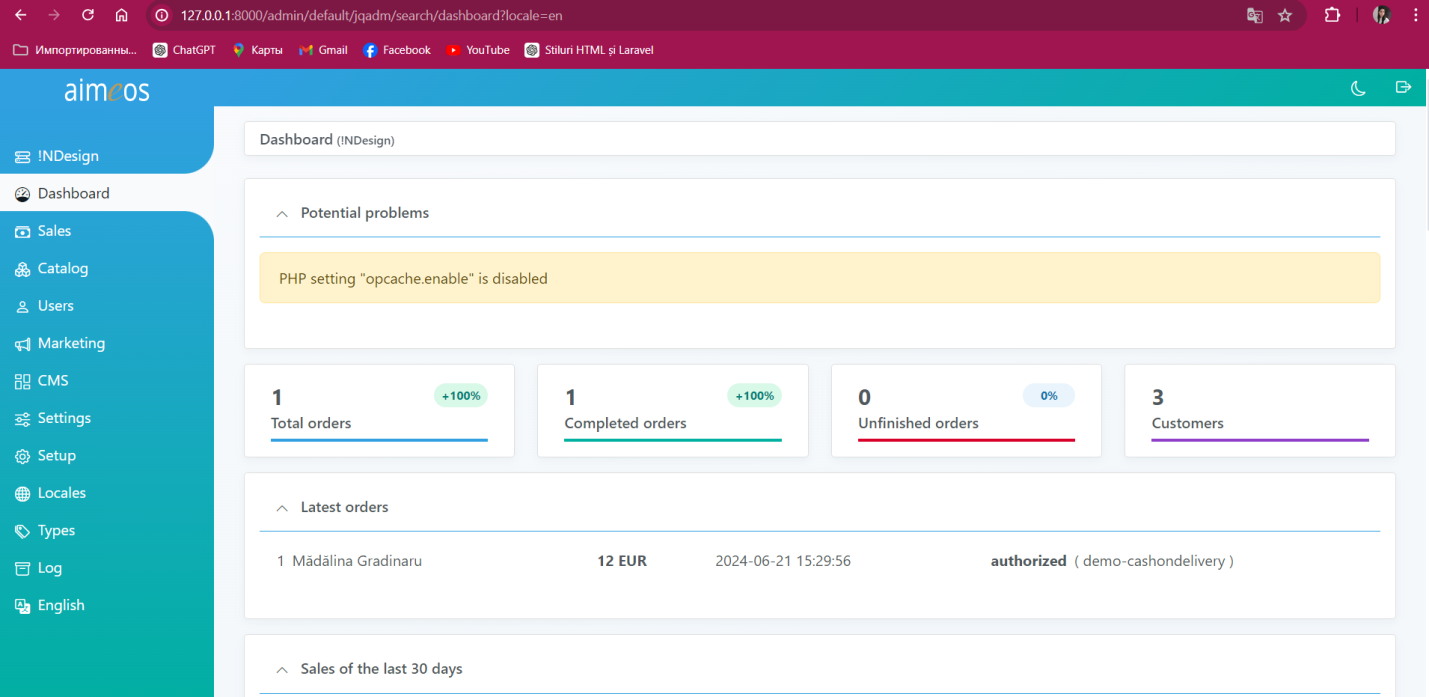
Un alt aspect esențial în configurarea inițială a magazinului online cu Aimeos este crearea unui cont de administrator , fapt solicitat in contextul instalării din linia de comandă .Acest pas asigură că doar utilizatorii autorizați pot accesa ulterior interfața de administrare și gestionare a magazinului.

Regula de accesare a panoului de administrator:

1. Înlocuirea adresei implicite a magazinului [http://127.0.0.1:8000/shop] cu

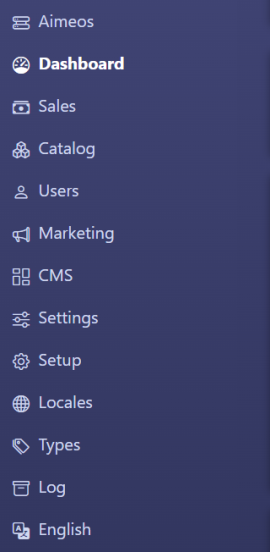
[http://127.0.0.1:8000/admin];

1. Navigarea spre formularul de logare în care se introduc datele solicitate la instalare:  
   
2. Afișarea panoului de gestionare a magazinului online:

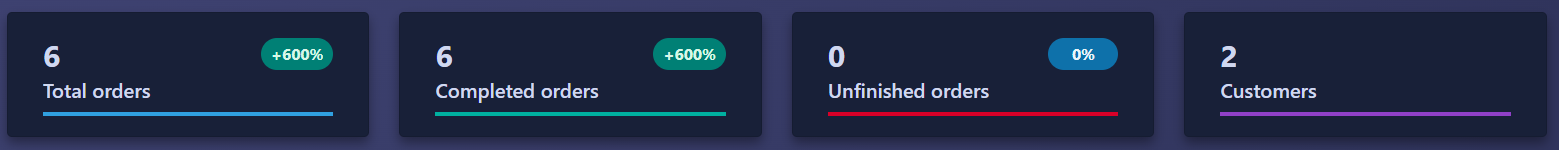


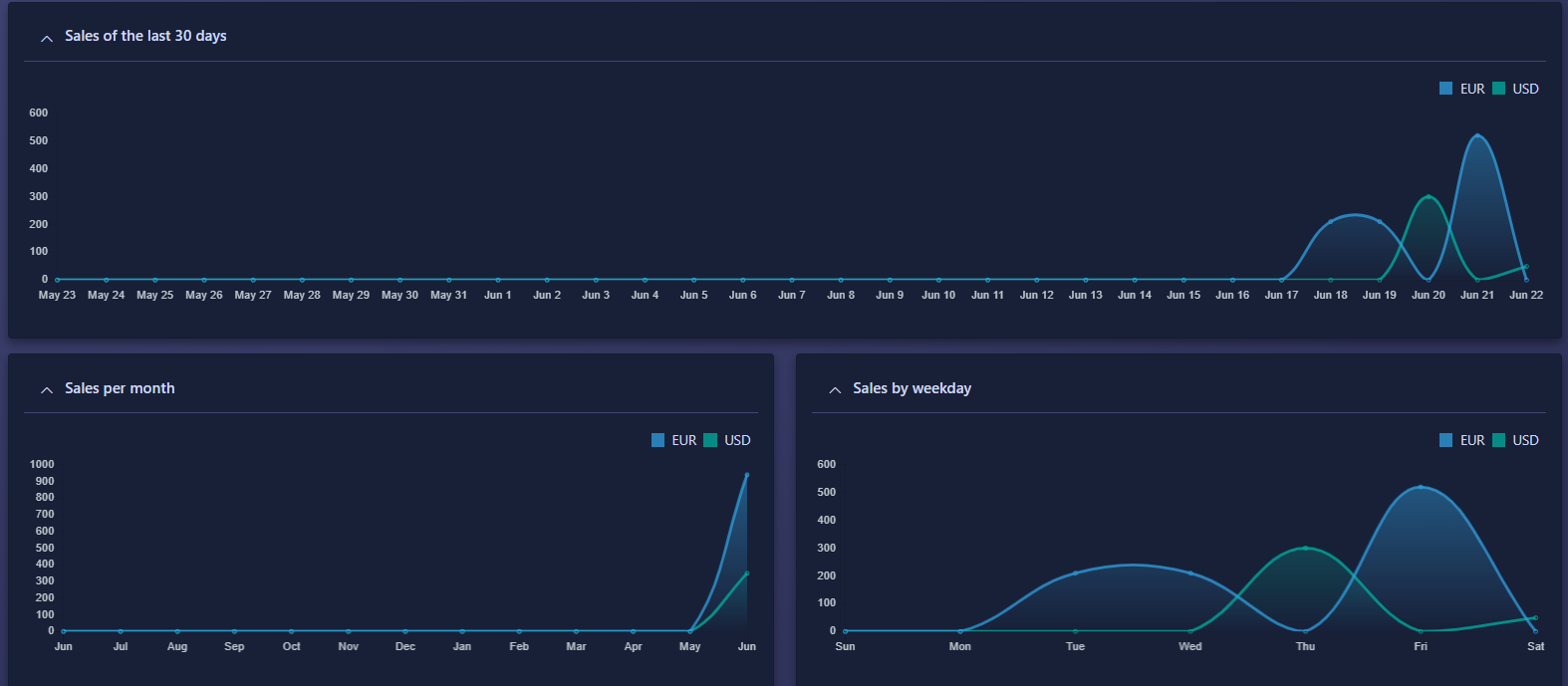
Panoul de administrare Aimeos oferă o interfață intuitivă și bine organizată pentru gestionarea magazinului online. Acesta este structurat pentru a permite acces rapid la toate funcționalitățile necesare pentru administrarea eficientă a unui magazin de comerț electronic.

1. Bara Laterală (Meniu Principal)

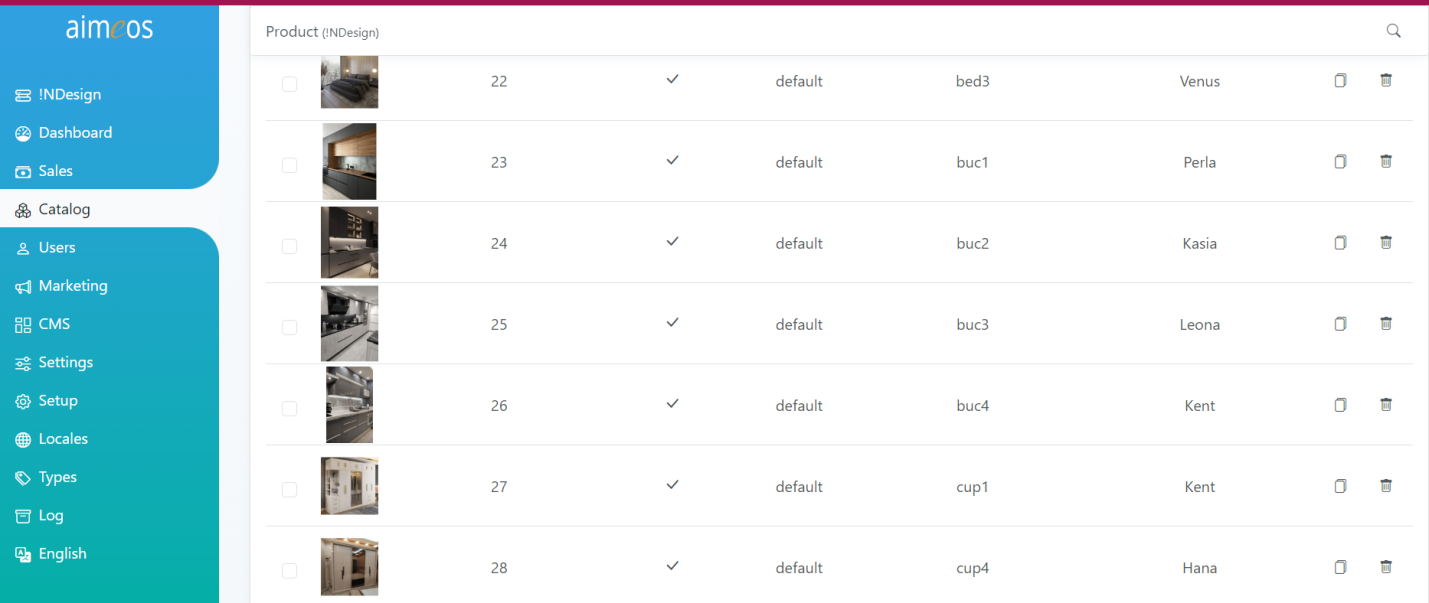


1. Administratorii pot vizualiza detaliile fiecărei comenzi, pot actualiza statusul comenzilor și pot gestiona retururile și rambursările:





1. Administratorii pot gestiona categoriile de produse, atributele și variantele prin adăugare, editare, ștergere.

****

Completând cele enumerate cu încă zeci de functionalități, se poate concluziona că Aimeos oferă o soluție completă, intuitivă și modernă pentru administrarea eficientă a unui magazin online cu elemente de decor interior **.**

# CONCLUZII

Proiectul de dezvoltare a site-ului web pentru compania !NDesign a evidențiat utilizarea eficientă a unor tehnologii moderne și instrumente specializate. Frontend-ul a fost implementat folosind Bootstrap, ceea ce a asigurat o interfață utilizator prietenoasă și adaptată dispozitivelor mobile. Caruselul dinamic de imagini, gestionat cu JavaScript, a adăugat interactivitate și a optimizat prezentarea proiectelor de design interior. Pe partea de backend, am folosit Laravel, beneficiind de framework-ul său robust pentru gestionarea eficientă a întregului sistem. Utilizarea Docker pentru containerizarea aplicației a simplificat gestionarea infrastructurii și a asigurat portabilitatea între diferitele medii de dezvoltare și producție. DBeaver a fost instrumentul principal pentru administrarea și gestionarea bazei de date PostgreSQL, asigurând performanță optimă și gestionarea eficientă a datelor.

Una dintre provocările majore a fost configurarea și integrarea eficientă a Laravel pentru optimizarea funcționalităților magazinului online. Prin implementarea acestuia, adițional și Aimeos, am reușit să dezvoltăm și să personalizăm cu succes caracteristici esențiale ale paginii de prezentare a companiei !Ndesign, precum și platformei de e-commerce, inclusiv gestionarea catalogului de produse și a coșului de cumpărături. Astfel, am creat o experiență de cumpărare fluentă și intuitivă pentru utilizatori.

De asemenea, am reușit să integram în mod securizat servicii externe pentru procesarea plăților, utilizând autentificarea OAuth și implementând un middleware personalizat pentru a asigura protecția datelor sensibile ale utilizatorilor. Aceste măsuri au consolidat securitatea și încrederea utilizatorilor în procesul de cumpărare online.Extinderea și îmbunătățirea site-ului web !NDesign poate aduce beneficii semnificative prin implementarea mai multor inițiative inovatoare și funcționalități noi:

* Adăugarea unor software-uri 3D avansate pentru modelare și vizualizare a design-urilor de interior poate transforma complet experiența utilizatorilor. Aceste instrumente ar permite vizualizarea în timp real a proiectelor propuse, permițând clienților să exploreze și să interacționeze virtual cu spațiile create de designeri. Aceasta nu numai că ar îmbunătăți experiența utilizatorului, dar ar și accelera procesul de luare a deciziilor și de aprobare a proiectelor.
* Prin adăugarea unor noi categorii de produse, precum mobilier personalizat, decorațiuni specifice și accesorii pentru diferite stiluri de interior, site-ul ar putea atrage o audiență mai largă și ar satisface mai bine diversitatea nevoilor clienților. O gamă extinsă de produse ar consolida poziția site-ului ca o destinație unică pentru toate necesitățile de design interior ale clienților.
* Implementarea unor strategii avansate de optimizare pentru motoarele de căutare (SEO) ar putea crește vizibilitatea site-ului în rezultatele căutărilor organice. Aceasta include optimizarea conținutului, structura site-ului, performanța paginilor și utilizarea cuvintelor cheie relevante pentru a atrage trafic de calitate și a stimula conversiile. O campanie SEO bine executată ar putea consolida poziția site-ului !NDesign în fața concurenței și ar aduce mai mulți clienți potențiali.
* Adăugarea unei funcționalități de chat în timp real pe site ar facilita comunicarea directă și interactivă între clienți și designerii de interior. Această facilitate ar permite clienților să primească consultanță imediată, să pună întrebări despre servicii sau produse, să solicite oferte personalizate și să primească feedback rapid și detaliat de la profesioniști. Un chat în timp real ar spori angajamentul utilizatorilor și ar crește încrederea în serviciile oferite de companie.
* Implementarea Realității Virtuale (VR): Permiterea utilizatorilor să experimenteze virtual proiectele de design interior prin tehnologia VR ar oferi o perspectivă realistă și interactivă asupra spațiilor propuse.

Aceste inițiative ar contribui semnificativ la consolidarea poziției site-ului web !NDesign ca o destinație de top în domeniul designului interior, oferind o experiență captivantă și interactivă pentru utilizatori și clienți.

În concluzie, proiectul !NDesign a demonstrat abilități avansate în gestionarea unor tehnologii complexe și în implementarea unor soluții personalizate pentru cerințele specifice ale unei companii de design interior. Utilizarea eficientă a Docker, DBeaver, PostgreSQL și Aimeos a contribuit semnificativ la succesul proiectului, oferind o platformă web robustă și scalabilă pentru promovarea și vânzarea serviciilor de design. Prin integrarea și gestionarea eficientă a acestor instrumente și tehnologii, proiectul a stabilit bazele pentru dezvoltarea viitoare și îmbunătățirea continuă a experienței utilizatorului și a performanței generale a site-ului web.

# REFERINȚE BIBLIOGRAFICE

[ ] (https://aimeos.org/docs/latest/developer/)

[ ] https://aimeos.org/docs/latest/developer/architecture/

[ ] <https://getcomposer.org/doc/00-intro.md>

[ ] <https://laravel.com/docs/11.x>