

**LUCRARE DE LICENȚĂ**

**Absolvent:** Bolba-Mateescu Andrei-Ioan

**Coordonator:** Vasilescu Anca

Brașov

Iunie-Iulie 2023

Cuprins

[1. Introducere 2](#_Toc133779552)

[1.1. Scurtă descriere a temei 2](#_Toc133779553)

[1.2. Motivația alegerii lucrării 2](#_Toc133779554)

[1.3. Contribuții personale 2](#_Toc133779555)

[1.4. Structura lucrării 2](#_Toc133779556)

[2. Abordarea temei 2](#_Toc133779557)

[2.1. Scopul lucrării 2](#_Toc133779558)

[2.2. Alte abordări în domeniu 2](#_Toc133779559)

[3. Managementul entităților 2](#_Toc133779560)

[3.1. Modelarea entităților 2](#_Toc133779561)

[3.2. Faza de implimentare a bazei de date 2](#_Toc133779562)

[3.3. Managementul identităților si al utilizatorilor 2](#_Toc133779563)

[4. Arhitectura aplicației 2](#_Toc133779564)

[4.1. Separarea preocupărilor 2](#_Toc133779565)

[4.2. Arhitectura REST pentri designul unui API 3](#_Toc133779566)

[4.3. Modelul “Clean Arhitecture” definit de Uncle Bob 3](#_Toc133779567)

[4.4. CRUD folosind verbe HTTP 3](#_Toc133779568)

[4.5. Securitate 3](#_Toc133779569)

[5. Noțiuni teoretice și tehnologii folosite 3](#_Toc133779570)

[5.1. .NET 7 3](#_Toc133779571)

[5.2. Angular 6](#_Toc133779572)

[5.3. Microsoft SQL 6](#_Toc133779573)

[5.4. Securitatea datelor 6](#_Toc133779574)

[6. Prezentarea aplicației 7](#_Toc133779575)

[6.1. Introducere 7](#_Toc133779576)

[6.2. Stuctura aplicației 7](#_Toc133779577)

[6.3. Persistența datelor 7](#_Toc133779578)

[7. Concluzii și perspective de viitor 7](#_Toc133779579)

[7.1. Concluzii 7](#_Toc133779580)

[7.2. Posibilități de extindere 7](#_Toc133779581)

[8. Bibligrafie 7](#_Toc133779582)

# Introducere

## Scurtă descriere a temei

## Motivația alegerii lucrării

## Contribuții personale

## Structura lucrării

# Abordarea temei

## Scopul lucrării

## Alte abordări în domeniu

# Managementul entităților

## Modelarea entităților

## Faza de implimentare a bazei de date

## Managementul identităților si al utilizatorilor

# Arhitectura aplicației

## Separarea preocupărilor

## Arhitectura REST pentri designul unui API

## Modelul “Clean Arhitecture” definit de Uncle Bob

## CRUD folosind verbe HTTP

## Securitate

# Noțiuni teoretice și tehnologii folosite

Aplicația InternHub este o aplicațit full-stack, din punct de vedere al dezvoltării, fiind formată din două parți: frontend și backend. Persistarea datelor a fost facută utilizând Microsoft SQ Server.

Pentru partea de backend a fost folosit framework-ul .NET cu C# ca si limbaj de programare. Motivarea pentru acesta alegere va fii în urmatoarele pagini, începând cu subcapitolul 5.1. pentru partea de backend.

Frontend-ul a fost realizat utilizând Angular și ca și limbaj de programare s-a folosit typescript. De asemenea motivarea pentru folosirea acestui framework o vom aborda la subcapitolul 5.2.

Pentru persistența datelor s-a folosit Microsoft SQ Server împreună cu Entity Framwrok pentru comunicarea backend-ului cu baza de date. Și pentru acesta avem motivarea alegerii la subacpitolul 5.3.

## .NET 7

.NET 7 este successoul lui .NET 6 și se focusează să fie unificat, modern, simple și de asemenea rapid. .NET 7 o să aibă suport tehnic pentru o perioadă de 18 luni, termen ce este standard in cadrul ecosistemului .NET. Lansat pe data de 8 noiembrie 2022, acesta va avea parte de suport tehnic până la data de 14 mai 2024.

Printre noile caracteristici aduce de la .NET 6 la .NET 7 se enumeră performanța, serializarea JSON, matematica generică, expresii regulate, noi librării .NET, sdk .NET și de asemenea actulizări ce rezolva unele probleme ale subsistemelor ce folosesc ecosistemul .NET.

Pentru a întelege mai în detaliu ce înseamnă aceste noi caracteristici pentru .NET 7, o să le dezvoltăm puțin pe unele dintre cele prezentare mai sus:

* **Performața**: Performanța este una dintre caracteristicile cheie ale .NET 7, iar toate caracteristicile sale sunt concepute ținând cont de performanță. În plus, .NET 7 include îmbunătățiri destinate exclusiv performanței pentru următoarele îmbunătățiri: înlocuirea pe stivă (On-stack replacement), optimizare ghidată pe profil (Profile-guided optimization), imbunatățirea generării de cod pentru Arm64 și performanța îmbunatățită pentru timpul de rulare mono
* **Serializarea JSON:** .NET 7 include îmbunătățiri ale serializării System.Text.Json în următoarele domenii: serializarea polimorfică și suport pentru membrii necesari
* **Matematica generică**: .NET 7 și C# 11 includ inovații care vă permit să efectuați operații matematice în mod generic.Scrierea unei funții care are ca scop efectuarea unor operații matematice pentru două numere se simplifică, deoarece nu mai este nevoie supraîncărcarea metodei pentru fiecare tip dorit. Acum este posibilă scrierea unei metode generice în care parametrul tip este constrâns să fie un tip asemănător unui număr.
* **Expresii regulate**: Biblioteca de expresii regulate a .NET a înregistrat îmbunătățiri semnificative funcționale și de performanță în .NET 7 [1]

Ecosistemul .Net este un ecosistem dezvoltat si menținut de către Microsoft care rulează în principal pe Microsoft Windows. Dezvoltarea acestiu ecosistem a început la finalul anilor 1990, sub alt nume față de cel cunoscut astăzi. Inițial numit Next Generation Windows Services, era parte a strategiei .NET. Odată cu începutul anului 2000, primele versiuni beta ale lui .NET 1.0 au fost lansate, urmând ca in februarie 2002 să fie anunțată și totodată lansată prima versiune a .NET, și anume .NET 1.0. În prezent framework-ul se alfă la versiunea .NET 8, urmând ca în noiembrie 2023 să fie lansata versiunea .NET 8.

Cel mai mare avantaj al framework-ului .NET este că acceptă platforma Windows. Aproape toată lumea lucrează cu mașini Windows. [2]

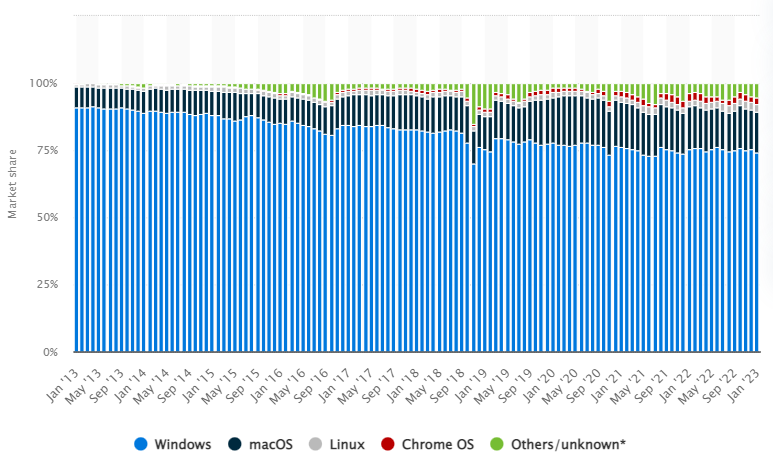


Figura 5.1 - Cota de piață globală deținută de sistemele de operare

Conform figurii 5.1 se poate observa că sistemul de operare preferat în întreaga lume este windows cu un procetaj constant mai mare de 74% în ultimii 10 ani. [3]

Un alt argument în favoarea ecosistemului .NET este constanța lansării de noi versiuni ale acestui ecosistem, și totodată menținerea suportului pentru versiunile lansate anterior versiunii curente, pentru a face mai ușoară trecerea de la o versiune la alta. La momentul actual acest timp de suport tehnic este de 6 luni după lansarea ultimei versiuni.

Pentru lucrarea în cauză am ales să lucrez cu .NET 7, deoarece este cea mai nouă și care are avantaje considerabile în fața altor opțiuni. Un alt motiv pentru .NET 7 este și incheierea suportului tehnic pentru .NET 3.1 în decembrie 2022, la puțin timp după lansarea lui .NET 7, iar tendința generală în cadrul companiilor este de a migra codul de la .NET 3.1 la .NET 6 sau .NET 7. Motivul pentru .NET 6 este că, asemenea lui .NET 3.1 este o versiune LTS (Long Term Suport), ceea ce îi oferă suport până în noiembrie 2024. [4]

Motivul pentru alegerea verisunii curent in detrimentul unei versunii LTS este, pe lângă cele prezentate la începutul acestui subcapitol este faptul că am dorit sa experimentez ultima versiune.

C# este un limbaj de programare care acceptă mai multe paradigme.Inițial lansat în 2000, acesta a ajuns deja la versiunea 11 lansată odată cu .NET 7 în noiembrie 2022. Asemănător lui .NET, C# primește contant actulizări și lansări de noi versiuni, astfel fiind o alegere optimă pentru partea de backend și totodată pentru frontend. [5]

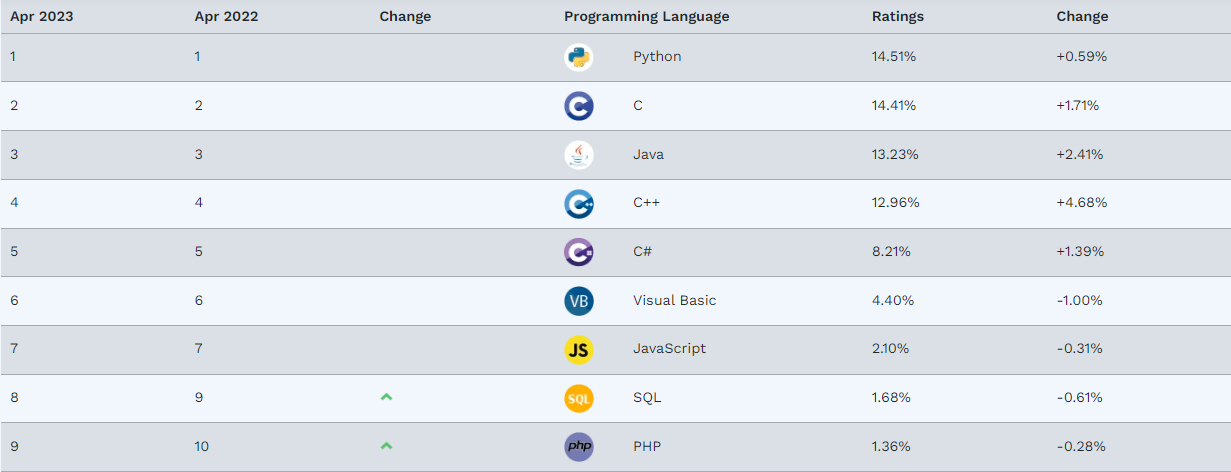


Figura 5.2 – TIOBE Programming Community Index

Conform figurii 5.2 se poate observa că deși C# nu este cel mai popular limbaj de programare este în continuă creștere fiin al 5-lea cel mai utilizat limbaj de programare. În iunie 2001, la puțin timp după lansare avea un raiting de 0,84%, iar este la 8,21% cu o creștere de 1.39% în ultimul an. [6]

Legat de API, pentru acesta s-a folosit ASP.NET Core Web API. Conform stasticilor făcute de cei la stack overflow din 2022 legată de framewrok-uri de dezvoltare pentru aplicatii web ASP.NET Core și ASP.NET sunt destul de populare fiind imediat după Angular și Vue.js în rândul tuturor celor intervievați, în număr de 58 743, așa cum se poate observa în figura 5.3. În rândul programatoril experimentați ASP.NET Core urcă o poziție deasupra lui Vue.js, iar ASP.NET rămâne pe acceți pozitie. Numărul programatoril experimentați intervievași de către cei de la Stackoverflow este puțin mai mic, dar rămâne un numâr consderabil de persoane intervievate, în număr de 45 297 , așa cum se poate observa în figura 5.4. În schimb cei ce sunt la început de drum aleg mai puțin ASP.NET și ASP.NET Core. Numărul celor intervievați ce se află la început de carieră este de 4932, așa cum se poate observa în figura 5.3. Cu toate acestea primele 2 opțini sunt la fel, fie ca vorbim de utilizator cu experința fie că vorbim de cei la început de drum. [7]

|  |  |
| --- | --- |
| Figura 5.3 – Toate răpunsurile la sondaj | Figura 5.4 – Răspunsurile programatorilor experimentați la sondaj |
| Figura 5.4 - Răspunsurile programatorilor aflați la început de carieră la sondaj |  |

## Angular

## Microsoft SQL

## Securitatea datelor

# Prezentarea aplicației

## Introducere

## Stuctura aplicației

## Persistența datelor

# Concluzii și perspective de viitor

## Concluzii

## Posibilități de extindere

# Bibligrafie

[1] What's new in .NET 7, <https://github.com/dotnet/docs/blob/main/docs/core/whats-new/dotnet-7.md>

[2] .NET, https://en.wikipedia.org/wiki/.NET

[3] Global market share held by operating systems for desktop PCs, from January 2013 to January 2023, <https://www.statista.com/statistics/218089/global-market-share-of-windows-7/>

[4] .NET and .NET Core Support Policy, <https://dotnet.microsoft.com/en-us/platform/support/policy/dotnet-core>

[5] C Sharp (programming language), <https://en.wikipedia.org/wiki/C_Sharp_(programming_language)>

[6] C# in 2023: The MOST POPULAR Programming Language?, <https://www.bytehide.com/blog/c-wants-to-become-the-most-popular-programming-language-in-2022>

[7] 2022 developer survey, <https://survey.stackoverflow.co/2022/#most-popular-technologies-webframe>