MINISTERUL EDUCAȚIEI NAȚIONALE

UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ŞI CALCULATOARE
DEPARTAMENTUL CALCULATOARE

BITSTORED
SISTEM DE STOCARE SECURIZATĂ A FIŞIERELOR

LUCRARE DE LICENŢĂ

Absolvent: **Diana BEJAN**

Conducător ştiinţific: **Senior Lector Eng. Cosmina IVAN**

2019

MINISTERUL EDUCAȚIEI NAȚIONALE

UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ŞI CALCULATOARE
DEPARTAMENTUL CALCULATOARE

DECAN,
Prof. dr. ing. Liviu MICLEA

DIRECTOR DEPARTAMENT,
Prof. dr. ing. Rodica POTOLEA

Absolvent: **Diana BEJAN**

BITSTORED
SISTEM DE STOCARE SECURIZATĂ A FIŞIERELOR

1. **Enunţul temei:** Crearea unui sistem de stocare a fisierelor in cloud, acesta fiind disponibil sub forma de mobile(IOS si Android) si de aplicaţie web. Aplicaţia realizazează stocarea fişierelor în forma criptată, pentru a oferi protecţie sporită a datelor, şi compresată pentru utilizarea eficientă a spaţiului de stocare al utilizatorului. De asemenea sistemul oferă un mecanism de restabilire a datelor printr-un sistem de logare avansat.

2. **Conţinutul lucrării:** Pagina de prezentare, Cuprins, Introducere, Obiectivele Proiectului, Studiu Bibliografic, Analiză şi Fundamentare Teoretică, Proiectare de Detaliu şi Implementare, Testare şi Validare, Manual de Intalare şi Utilizare, Concluzii ,Bibliografie, Anexe.

3. **Locul documentării:** Universitatea Tehnică din Cluj-Napoca, Departamentul Calculatoare

4. **Consultanţi:** Senior Lector Eng. Cosmina Ivan

5. **Data emiterii temei:** 1 ianuarie 2019

6. **Data predării:** 12 iulie 2019

Absolvent: _____

Coordonator ştiinţific: _____

MINISTERUL EDUCAȚIEI NAȚIONALE

UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA

FACULTATEA DE AUTOMATICĂ ŞI CALCULATOARE
DEPARTAMENTUL CALCULATOARE

Declaraţie pe proprie răspundere privind autenticitatea lucrării de licenţă

Subsemnatul(a) _____, legitimat(ă) cu _____ seria _____ nr. _____
CNP _____, autorul lucrării _____
_____
_____
elaborată în vederea susţinerii examenului de finalizare a studiilor de licenţă la Facultatea de Automatică şi Calculatoare, Specializarea _____ din cadrul Universităţii Tehnice din Cluj-Napoca, sesiunea _____ a anului universitar _____, declar pe proprie răspundere, că această lucrare este rezultatul propriei activităţi intelectuale, pe baza cercetărilor mele şi pe baza informaţiilor obţinute din surse care au fost citate, în textul lucrării şi în bibliografie.

Declar, că această lucrare nu conţine porţiuni plagiate, iar sursele bibliografice au fost folosite cu respectarea legislaţiei române şi a convenţiilor internaţionale privind drepturile de autor.

Declar, de asemenea, că această lucrare nu a mai fost prezentată în faţa unei alte comisii de examen de licenţă.

În cazul constatării ulterioare a unor declaraţii false, voi suporta sancţiunile administrative, respectiv, *anularea examenului de licenţă*.

Data _____

Nume, Prenume _____

Semnătura

Cuprins

Capitolul 1 Introducere - Contextul proiectului **4**

Capitolul 2 Obiectivele Proiectului **7**
2.1 Formularea temei 7
2.2 Obiectivele proiectului 9
2.3 Cerinţe 9
2.3.1 Cerinţe funcţionale 9
2.3.2 Cerinţe non-funcţionale 11

Capitolul 3 Studiu Bibliografic **12**
3.1 Caracteristicile arhitecturii monolitice 12
3.2 Caracteristicile şi avantajele arhitecturii orientate pe microservicii 14
3.3 Microservicii în Cloud 16
3.4 Securitatea in sistemele informatice 17
3.4.1 Ameninţări 18
3.4.2 Criptografia 19
3.4.3 Compresia 19
3.4.4 Steganografia 19
3.5 Sisteme similare 19
3.5.1 Metodologia de analiză 20
3.5.2 CloudMe 21
3.5.3 Dropbox 22
3.5.4 CrashPlan 24
3.5.5 ICloud 25
3.5.6 Google Drive 27
3.5.7 OneDrive 28
3.5.8 pCloud 29
3.5.9 sync.com 30
3.5.10 Concluzii şi plasarea sistemului 31

Capitolul 4 Analiză şi Fundamentare Teoretică **33**
4.1 Cazuri de utilizare . . . 34
4.1.1 Actori . . . 35
4.1.2 Modele de cazuri de utilizare . . . 35
4.2 Arhitectura canceptuală a sistemului . . . 35
4.3 Tehnologii . . . 35
4.3.1 Golang . . . 35
4.3.2 gRPC . . . 36
4.3.3 VueJS . . . 36
4.3.4 MongoDB . . . 36
4.3.5 Couchbase . . . 36
4.3.6 HTML, CSS, Bootstrap . . . 36
4.3.7 JSON Web Token(JWT) . . . 36
4.3.8 Docker şi Kubernetes . . . 36
4.3.9 Google Cloud . . . 36
4.3.10 Git . . . 36

Capitolul 5 Proiectare de Detaliu şi Implementare **37**
5.1 Arhitectura serverului . . . 37
5.1.1 Descriere generală . . . 37
5.1.2 Orchestrarea microserviciilor . . . 39
5.1.3 Microserviciul de autentificare . . . 39
5.1.4 Microserviciul de criptare . . . 40
5.1.5 Microserviciul de steganografie şi marcare . . . 40
5.1.6 Microserviciul de compresie . . . 41
5.1.7 Microserviciul de fisiere . . . 45
5.1.8 Microserviciul de utilizatori . . . 45
5.2 Arhitectura aplicatiei web . . . 45
5.2.1 Descriere generală . . . 45
5.2.2 Descrierea componentelor . . . 45

Capitolul 6 Testare şi Validare **46**
6.1 Testarea serverului . . . 46
6.1.1 Reguli de testare în Golang . . . 46
6.1.2 Testarea serviciilor . . . 47
6.2 Testarea clientului . . . 48

Capitolul 7 Manual de Instalare şi Utilizare **49**
7.1 Cerinţe preliminare . . . 49
7.2 Instalare si configurare . . . 49

Capitolul 8 Concluzii **50**
8.1 Contribuţii şi rezultate obţinute 50
8.2 Dezvoltări ulterioare 50

Bibliografie **51**

Anexa A Secţiuni relevante din cod **52**

Lista figurilor **53**

Lista tabelelor **54**

Anexa B Diagrame UML **55**

Anexa C Glosar **56**

Capitolul 1

Introducere - Contextul proiectului

În epoca contemporană se observă o tendinţă continuă a digitalizării si transformării digitale, fapt care aduce un impact enorm atât asupra marilor companii, atât şi asupra utilizatorilor individuali. Lumea bazată pe date va fi permanentă, mereu în urmărire, mereu în stadiu de monitorizare - pentru că va fi mereu în stadiu de învăţare.

IDC[1] a definit trei locaţii principale în care digitalizarea are loc şi unde este creat conţinutul de date: tip nucleu (centre de date tradiţionale si de tip cloud), tip muchie (infrastructuri de tip sucursală), şi obiectivele finale (PC-uri, telefoane si dispozitive IoT). Sumarizarea tuturor acestor date, în momentul în care sunt create, capturate sau replicate, se numeşte Global Datasphere, şi aceasta se confruntă cu o creştere spectaculoasă. IDC (International Data Corporation) estimează că volumul de date din Global Datasphere va creşte de la 33 Zettabytes[1] în 2018 până la 175 Zettabytes in 2025, evoluţia se poate observa în Figura 1.1.

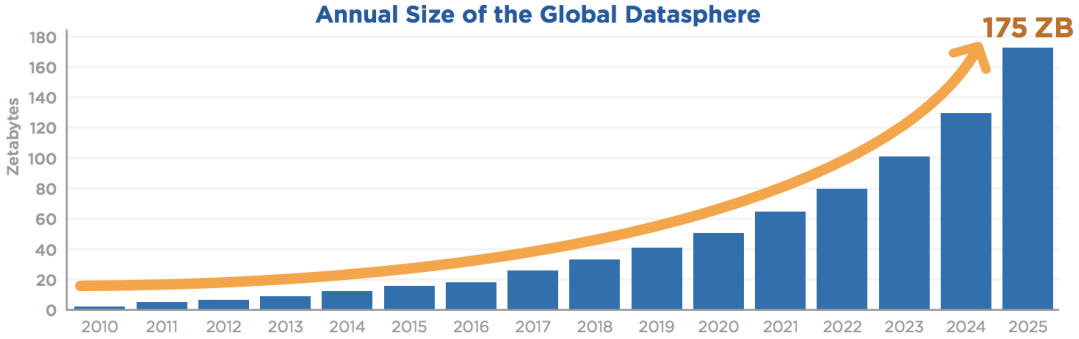

Figura 1.1: Creşterea volumului de date 2010-2025

În trecutul recent utilizatorii erau responsabili pentru datele lor, însa dependenţa şi încrederea lor în servciile cloud, în special din cauza conectivităţii, performanţei si con-

[1] 1 Zetta byte echivalent cu 2^{70} bytes

fortului, continua sa crească ceea ce duce la noi provocări pentru furnizorii de servicii cloud. Mediul afarcerilor urmăreşte centralizarea managementului datelor, pentru a putea oferi securitate, analiză de date, experienţă utilizator mai bună (prin comunicare între dispozitive, IoT, personalizarea profilului). Responsabilitatea pentru managementul datelor utilizatorilor şi businessurilor duce la o crestere continuă a centrelor de date ale furnizorilor de servicii Cloud. Ca rezultat importanţa serviciilor cloud creşte considerabil, iar utilizatorii nu doar permit acest lucru si se aşteaptă la o crestere cât mai spectaculoasă.

Volumul de date cât mai mare stocat în cloud este scopul industriei de stocare a datelor. Pentru a supravieţui într-o lume care tinde a fi condusă de inteligenţa artificială şi sistemel autonome sistemele de cloud au nevoie să se perfecţioneze tot mai mult pentru a ţine pasul cu evoluţia lumii.

În calitate de clienţi, oamenii doresc să obţină acces rapid şi simplu la datele lor indiferent de oră şi locaţia în care se află. Sistemele cloud sunt provocate să ofere servicii performante de acces, care nu vor expune datele utilizatorilor.

Organizaţiile şi utilizatorii au început să îşi schimbe destinaţia datelor de la infrastructuri fizice spre *cloud*-ul public, alţii însă au început sa îsi dezvolte propriile soluţii de stocare, astfel profitând de toate beneficiile *cloud*-ului. Însă securitatea datelor rămâne cea mai mare problemă, mai ales din cauza lipsei de control asupra infrastructurii fizice[2]. Toate sistemele încercă să se bazeze pe modelul CIA[2], acest sistem este prezentat în figura 1.2.

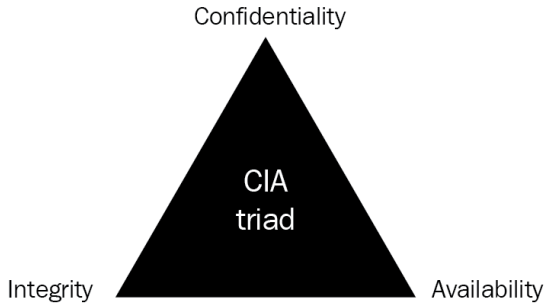

Figura 1.2: Triada CIA

Confidenţialitatea se referă la protejarea datelor de la un access neautorizat. Integritatea se referă la protecţia datelor de la modificări neautizate, orice modificare poate însemna o pierdere considerabilă pentru utilizator sau organizaţie. Disponibilitatea denotă faptul că informaţiile vor fi accesibile doar pentru utilizatorii care au drept de acces, o încălcare a acestei reguli, din nou, se va provoca pierderea unor date. Toate cele 3 aspecte sunt esenţiale pentru securitatea datelor, însă acestea sunt uneori complicat de oferit.

[2]Confidentiality, Integrity, and Availability

Securitatea datelor este o problemă foarte mare cu care se luptă zilnic dezvoltatorii de servicii *cloud*. Multe din probleme se datorează expunerii nivelului de stocare, datele ar trebui să fie stocate în medii securizate, care oferă criptare eficientă. Aceste problemepot fi înlăturate doar prin indeplinirea strictă a unor tratate de securitate avansată.

Creşterea volumului de date, pe de altă parte duce şi la creşterea volumului fizic de *hardware* necesar. Compresia datelor este o operaţie care permite stocarea aceluiaşi volum de date la un preţ mult mai redus[**?**]. Datorită compresiei un utilizator poate stoca un volum mai mare de date decât ii permite teoretic *hardware*-ul. Pe lânga faptul că se reduce spaţiul de stocare a datelor, compresia contribuie şi la micşorarea timpului necesar pentru transmisie sau pentru operaţiile I/E[**?**].

Un alt beneficiu al compresiei este creşterea nivelului de securitate. Acest fapt se datorează alterării datelor, astfel chiar şi un algoritm de criptare primitiv devine mult mai complicat de spart prin forţă brută, în ecuaţie se mai adaugă şi decompresia datelor, care este o operaţie costisitoare, nu se mai pot determina asocieri între simboluri, între secvenţe de caractere şi cuvintele dintr-o limbă. Cu toate că se adaugă un timp mare de procesare, în cazul unor date sensibile, această întârziere este una motivată din punctul de vedere al securităţii datelor.

Creşterea încrederii în sistemele de stocare, creşterea volumului de date, inclusiv şi a celor de sensibilitate înaltă, şi a numărului de atacuri cibernetice crează o nouă provocare pentru sistemele de stocare cloud existente: creşterea securităţii datelor. De asemenea acest aspect crează şi oportunităţi noi pentru sistemele la început de drum, le oferă o piaţă de defacere enormă şi un număr ridicat de clienţi care sunt gata să plătească preturi relativ ridicate pentru securitatea datelor sale.

Volumul de date creşte, însă pe lângă conceptul celor *trei V*: volum, varietate şi velocitate, se mai adaugă şi al 4-lea V: valoare. Pentru oferirea securităţii datelor, anual se cheltuie în jur de $100 miliarde, în 2019 această sumă a ajuns la $124 miliarde.

Atenţia industriei IT ar trebui să se concentreze în jurul securităţii cibernetice, cu referire mai mult la valoarea datelor, nu la volumul sau sursa lor. Soluţiile sunt şi trebuie să fie conduse de viziunea şi gândirea oamenilor, dar validarea soluţiilor ar trebui să fie validate de inteligenţa datelor.

Aceste fapte sunt un avantaj competitiv pentru indrustria IT şi sunt o provocare pentru construirea unei culturi sănătoase a datelor.

Capitolul 2

Obiectivele Proiectului

În acest capitol este prezentată tema proiectului, obiectivele şi cerinţele funcţionale esenţiale ale proiectului.

2.1 Formularea temei

Prin acest poroiect, se urmăreşte implementarea unei platforme de *Cloud Storage*, aceasta trebuie să ofere o bună protecţie a datelor, aceste date să nu fie accesibile dezvoltatorilor sistemului, atacatorilor sau oricărei alte instituţii care colectează date. Sistemul este menit pentru stocarea datelor sensibile, sub răspunderea directă a utilizatorilor, de asemenea sistemul poate fi privit ca o platformă fără cunoştinţe despre datele stocate, nimeni altul decât proprietarul fişierelor nu poate decripta datele.

Conceptul sistemului este bazat pe arhitectura clasică *client-server*, aceasta este prezentată în figura 2.1.

Figura 2.1: Arhitectura Client-Server

Serverul la rândul său având o arhitectură bazată pe microservicii, fiecare mcroserviciu fiind organizat sub formă de *layer*. La nivel de aplicaţie, responsabilităţile sunt partiţionate între server şi client. La nivelul serverului, funcţionalităţile esenţiale sunt repar-

tizate între servicii, astfel fiind respectat principiul **Single-responsability** din SOLID[**?**]. La nivel de microserviciu funcţionalităţile sunt divizate la nivel de *layer* şi funcţie.

Serverul se ocupă de prelucrarea şi stocarea datelor. Funcţionalităţile cheie sunt: compresie, criptare, conversie, steganografie, monitorizare şi stocare. Partea de stocare a datelor este împărţită în 2: fişiere şi date personale. Datele utilizator sunt stocate îin MongoDB, iar cele despre fişiiere in Couchbase, este imposibil să asociezi un fişier cu identiitatea unui utilizator fără accesul la ambele baze de date.

Aplicaţia va trebui să fie proictată astfel încât să poată satisface cereri de la sute sau mii de utilizatori concomitent, fără a implica întârzieri mari de răspuns. Apliicaţia va trebui să fie rezistenţă la un nivel mare de utilizare şi să se autoscaleze prin crearea unor replici ale aceluiaşi cod, doar pentru microserviciile utilizate intens.

Sistemul va oferi utilizatorilor funcţionalităţile de bază a unui sistem de stocare în *cloud*: încărcare fişier, descărcare fişier, creare fişier nou, grupare fişiere şi management fişiere. Pe lângă funcţionalităţile enumerate, sistemul va oferi un nivel de securitate înalt prin criptarea tuturor datelor şi eficienţă de utilizare a spaţiului prin compresia datelor. Cheia de criptare/decriptare nu va fi stocată nicăieri în server sau client, utilizatorul va ave în totalitate responsabilitatea de a îsi păstra fişierele sigure şi de a putea recupera datele stocate îin sistem, din păcate este imposibil să i se ofere asisitenţă în cazul pierderii parolei, regenerarea ei priin forţă brută ar putea dura între câteva ore şi câţiva ani.

De asemenea la descărcarea unui fişier, utilizatorul va avea posibilitatea de a ascunde mesaje în imagini, sau de a aplica marcaje vizibile pe imagini. Această funcţionalitate poate fi folosită atunci când se partajează un fişier, pentru a determina dacă datele au fost expuse diin sistem şi identiitatea celui care le-a expus. Un exemplu de steganografie pe imagini este prezentat în figura 2.2.

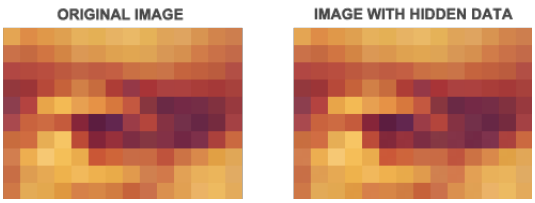

Figura 2.2: Exemplu de steganografie pe imagini

După cum se poate observa, manipularea biţilor nu este vizibilă pentru ochiul uman, însă pentru calculator poate conţine informaţii valoroase.

Pentru client, se urmăreşte implementarea unei aplicaţii web. Aceasta va trebui să interacţioneze cu utilizatorii prin interfaţa web şi cu serverul prin apeluri de tip gRPC, pentru a asigura o viteză mai bună de transmisie şi răspuns. Clientului nu îi vor fi cunoscute decât 3 dintre serviciile existente: serviciul de autentificare, servicul de management utilizatori şi serviciiul de management fişiere.

2.2 Obiectivele proiectului

- Sistemul va oferi utilizatorilor experienţa deplină a unui sistem de stocare în *Cloud*: încărcare fişier, descărcare fişier, modificare fişier, grupare fişiere.
- Sistemul va oferi utilizatorilor un nivel înalt de securitate a datelor personale prin folosirea unor algoritmi de criptare eficienţi. Cheile nu vor fi stocate în sistem, iar datele vor fi aproape imposibil de decriptat de către atacatori.
- Conexiunea între client şi server va fi securizată utilizând protocolul **HTTPS**. Astfel se va asigura că datele transmise vor fi protejate împotriva atcurilor de tipul *man-in-the-middle*.
- Folosirea spaţiiului de stocare va fi eficientizaţa prin folosirea unor algoritmii de compresie eficienţi, care permit reducerea volumului de date de maii mult de 2 ori în cazul unor date uzuale.

2.3 Cerinţe

În această secţiune sunt descrise şi enumerate cerinţele principale ale sstemului. Acestea se referă atât la funcţionalităţile propriu-zise, cât şi la experienţa utilizatorului.

2.3.1 Cerinţe funcţionale

Cerinţele funcţionale definesc funcţiile sistemului sau comportamentul său, unde funcţiile sunt descrise ca o specificaţie sau set de acţiuni dintre intrare şi ieşire. Pornind de la obiectivele proiectului, pot fi determinate următoarele cerinţe funcţionale:

CF1 Ca utilizator neînregistrat, doresc să îmi pot crea un cont nou.

CF2 Ca utilizator, doresc să mă autentific în sistem utilizând numele de utilizator şi parola, care au fost indicate la momentul înregistrării.

CF3 Ca Utilizator, doresc ca sesiunea mea să fie păstrată şi după închiderea *browser*-ului.

CF4 Ca utilizator, dorec să îmi opresc sesiunea curentă prin deautentificare.

CF5 Ca utilizator, dorec să am posibilitatea de a îmi bloca temporar contul, fără a pierde accesul la date.

CF6 Ca utilizator, dorec să am posibilitatea de a îmi debloca contul.

CF7 Ca utilizator, dorec să am posibilitatea de a îm sterge contul şi toate datele stocate.

CF8 Ca utilizator, doresc să pot modifica datele mele de utilizator, cum ar fi parola, nume, poză de profil.

CF9 Ca utilizator, doresc să pot primi o confirmare a înregistrării pe mail-ul indicat la momentul înregistrării în sistem.

CF10 Ca utilizator, doresc să pot vizualiza profilul meu.

CF11 Ca utilizator, doresc să pot vizualiza fişierele mele din *Drive* şi să pot naviga prin ierarhia de directoare.

CF12 Ca utilizator, doresc să pot crea un fişier nou, care să aibă sau nu conţinut.

CF13 Ca utilizator, doresc să pot crea un director nou.

CF14 Ca utilizator, doresc să pot încarca un fişier în sistem.

CF15 Ca utilizator, doresc să pot modifica un fişier care se află în spaţiul meu de stocare.

CF16 Ca utilizator, doresc să pot şterge un fişier din sistemul meu de fişiere.

CF17 Ca utilizator, doresc să pot schimba directorul în care se află un fişier.

CF18 Ca utilizator, doresc să pot crea o copie a unui fişier într-un alt director, fără a îm crşte spaţiul de stocare utilizat.

CF19 Ca utilizator, doresc să pot descărca un fişier care se află în spaţiul meu de stocare.

CF20 Ca utilizator, doresc să pot codifica mesaje în fişierul descărcat.

CF21 Ca utilizator, doresc să pot vedea dacă în fişierul pe care l-am încărcat sunt codificate mesaje.

CF22 Ca utilizator, doresc să mi se ofere date privind volumul de stocare economisit datorită funcţionalităţilor sistemului.

CF23 Ca utilizator, doresc să pot aplica mesaje vizuale pe imaginile descărcate.

CF24 Ca administrator, doresc să pot vizualiza conturile tuturor utilizatorilor.

CF25 Ca administrator, doresc să pot bloca contul oricărui utilizator.

CF26 Ca administrator, doresc să pot debloza contul oricărui utilizator.

2.3.2 Cerinţe non-funcţionale

Cerinţele non-funcţionale se referă mai mult la calitatea şi experienţa de utilizarea a sistemului, decât la functiionalităţile şi capabilităţile specifice. Aceste cerinţe descriu cum ar trebui să fie sistemul, nu ce ar trebui să facă acesta.

CNF1 Codul sursă al sistemului ar trebui să fie scris şi menţinut la cel mai înalt nivel.

(1) Codul trebuie să respecte principiile SOLID.

(2) Dependenţele la librării trebuie înnoite frecvent, pentru a asigura fixarea eventualelor probleme din versiunile precedent.

(3) Codul ar trebui să fie bine testat, cu o acoperire mai mare de 80%.

(4) Nu ar trebui să se folosească soluţii copiate de pe fodumuri de programare, acestea ar putea conţine vulnerabilităţi.

(5) Codul trebuie să fie bine documentat şi uşor de citit şi modificat.

CNF2 Sistemul trebuie să ofere un nivel înalt de securitate a datelor.

(1) Datele personale ale utilizatorilor vor fi stocate într-un mediu securizat, acestea vor fi criptate în avans şi vor fi decriptate doar la cererea proprietarului legitim.

(2) Fişierele vor fi criptate utilizând algoritmi compecşi, iar datele despre cheile de criptare nu vor fi stocate în sistem.

(3) Nu se va face o asociere directă între identitatea utilizatorului şi datele stocate în sistemul de fişiere.

CNF3 Sistemul trebuie să utilizeze eficient spaţiul de stocare.

(1) Datele vor fi compimate cu ajutorul unor algoritmi ce au o rată de compresie foarte mare.

(2) Nu se vor stoca date redundante sau duplicate.

CNF4 Sistemul va avea o interfaţă utilizator inteligibilă, uşor de utilizat şi care nu creează ambiguitate în modul de utilizare.

CNF5 Sistemul va fi rezistent la căderile unor anumite servicii şi îsi va putea reveni ulterior, fără ca utilizatorul şa ştie acest lucru.

CNF6 Clientul trebuie să fie suportat de mai multe browsere şi să ofere aceeaşi interfaţă web.

Capitolul 3

Studiu Bibliografic

3.1 Caracteristicile arhitecturii monolitice

Monolit însemnă ”dintr-o bucată”. O aplicaţie monolitică este o aplicaţie software în care diferite componente au fost combinate într-un singur program. Componentele programului sunt interconectate şi interdependente, spre deosebire de abordarile modulare care oferă un nivel de cuplare scăzut. Pentru ca programul să fie compilat sau executat fiecare componentă trebuie să fie prezentă şi definită şi să existe legăturile cu fiecare componentă. Componentele aplicaţiei pot fi:

- Autorizarea - responsabilă pentru autoriazarea utilizatorului.
- Prezentarea - responsabilă pentru tratarea apelurilor HTTP şi servirea răspunsurilor.
- Logica de business.
- Componenta de acces la baza de date.

Un monolit poate fi considerat un pattern arhitectural sau un stil de dezvoltare a aplicaţiilor (sau un anti-pattern, dacă privim din perspectiva dezavantajelor). Stilurilot şi pettern-urile sunt de obicei grupate în categorii sau seturi, pentru a fi mai uşor de asociat. Categoriile de bază pentru arhitectura monolitică sunt:

- Modul - unităţile de cod sunt separate în module şi sunt compilate împreună producând un singur artefact.
- Alocare - Toate componentele sistemului sunt compilate, livrate şi configurate în acelaşi timp, ca un singur artefact, toate având aceeaşi versiune, indifernt de câte ori au fost modificate. Numărul versiunii este egal cu numărul de livrari al artefactului
- Runtime - Există o singură instanţa aplicaţiei care execută toate sarcinile.

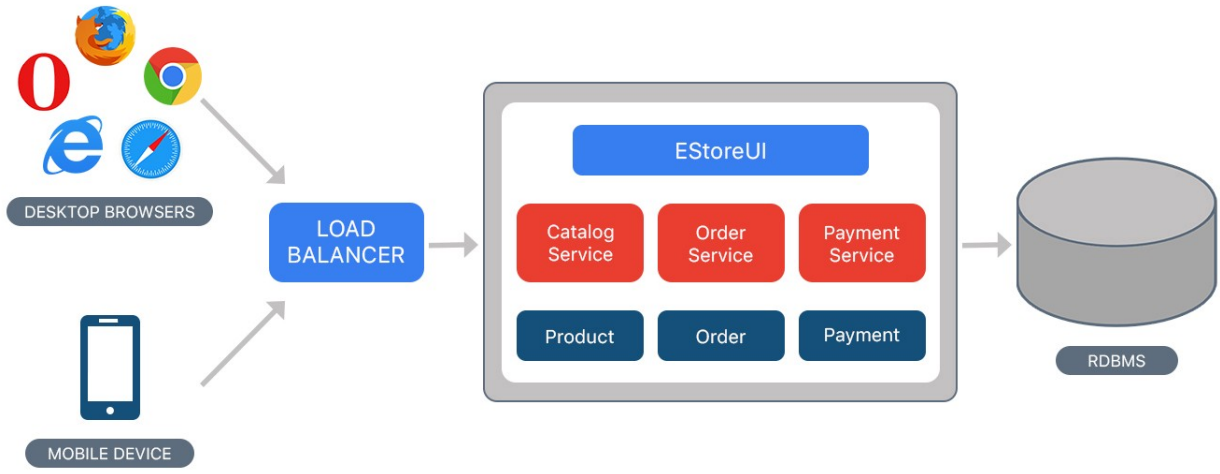

Figura 3.1: Arhitectura monolitică

Avantajele arhitecturiii monolitice sunt:

- Uşor de dezvoltat - la începutul unui proiect este mult mai uşor să dezvolţi o arhitectură monolitică.
- Uşor de testat. De exemplu, se pot implementa teste end-to-end prin simpla rularea a aplicaţiei şi testarea ei cu un tool specializat.
- Uşor de pus în funcţiune, este necesară doar copierea pe un server şi rularea programului.
- Scalabilă pe orizontală prin rularea mai multor instanţe.

Arhitectura monolitică este abordarea tradiţională care este folosită în multe sisteme, care sunt construite ca o aplicaţie autonomă. Chiar dacă este prezentă în multe aplicaţii existente şi încă este folosită pentru dezvoltarea aplicaţiilor noi, limitările şi problemele existente în acest mod de abordare duc la creşterea popularităţii arhitecturii bazate pe microservicii. **Dezavantajele** arhitecturii monolitice sunt:

- Mentenanţa - dacă o aplicaţie este prea mare este foarte greu să faci schimbări rapide şi să nu afectezi funcţionarea corectă a altor componente.
- Dimensiunea aplicaţiei duce creşterea timpul de start-up.
- Toată aplicaţia va trebui restartată atunci când se face o schimbare de cod.
- Este foarte complicat de scalat.
- O problemă în una dintree componente poate afecta întreaga aplicaţie.

- Este complicat să adopte tehnologii si framework-uri noi, deoarece prea multe lucruri trenuie schimbate în acelaşi timp.
- Spre finalul ciclului de dezvoltare complexitatea de a scrie cod devine mai mare, iar raportul timp-eficienţă devine tot mai mare.

Această arhitectură are şi plusuri şi minusuri, dar, dat fiind faptul că de fiecare dată când se rescrie o porţiune de cod este necesară recompilarea întregului program, arhitectura monolitică duce la întârzieri destul de mari, cauzate de compilările repetate a întregului program.

3.2 Caracteristicile şi avantajele arhitecturii orientate pe microservicii

Microserviciile sunt nişte entităţi mici, independente, autonome, create să funcţioneze împreună, fiecare dintre ele este focusat pe un singur lucru şi are scopul de a-l face bine. Arhitectura bazată pe microservicii este un stil arhitectural care structurează aplicaţia ca o colecţie de servicii indepenedente şi modulare, care sunt uşor de testat, de întreţinut şi de înţeles. Acest tip de abordare duce la creşterea agilităţii prin îmunătăţirea productivităţii şi scăderea timpului de dezvoltare a produsului. Microserviciile au demonstrat că sunt un sistem de nivel superior, în special pentru aplicaţii mari care sunt dezvoltate de mai multe echipe. Pe lângă beneficiile enumerate mai sus, microserviciile mai oferă următoareele avantaje:

- Sunt mentenabile.
- Sunt scalabile.
- Sunt uşor de testat.
- Au nivel de cuplare joasă.
- Sunt independente.
- Sunt rezistente la căderi.
- Sunt organizate în funcţie de capabilităţile şi funcţionalităţile de business.
- Oferă independenţă dezvoltatorilor.
- Pot fi dezvoltate independent.
- Modificările pot fi aplicate uşor, deoarece nu mai necesită recompilarea întregului produs.

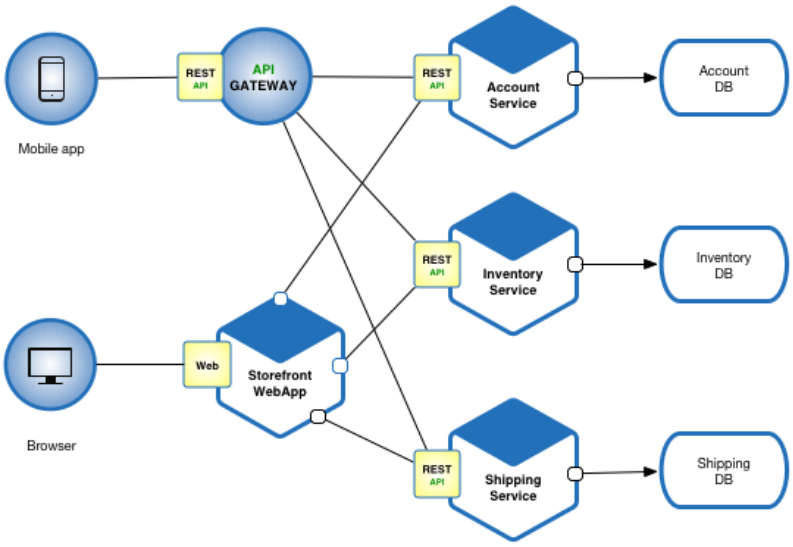

Figura 3.2: Arhitectura bazată pe microservicii

Arhitectura bazată pe microservicii permite integrarea şi livrarea continuă a unui produs complex şi de volum mare, deasemnea promovează diversitatea tehnologiilor utilizate într-un proiect. În prezent, tot mai multe companii au început să folosească arhitecturi bazate pe microservicii pentru produsele lor. Câteva dintre aceste companii sunt[3]:

- Netflix
- eBay
- Amazon
- Twitter
- PayPal
- SoundCloud
- Gilt
- The Guardian

Netflix, eBay şi Amazon sunt cunoscute pentru arhitecturile lor diverse, care au evoluat de la *Monolit* la *Microservicii* cu scopul de a putea face faţă unor volume imense de date.

Totuşi, ca oricare altă soluţie, arhitectura bazată pe microservicii are o serie de dezavantaje[4]:

- Se adaugă complexitate din cauza creării unor sisteme distribuite.

- Programatorul trebuie să implementeze comunicarea între servicii.
- Testarea interacţiunii este destul de complexă.
- IDE-urile şi tool-urile existente au un număr scăzut de funcţionalităţi care ajută la dezvoltarea aplicaţiilor distribuite.
- Un sistem format din microservicii are un nivel mai ridicat al consumului de resurse.

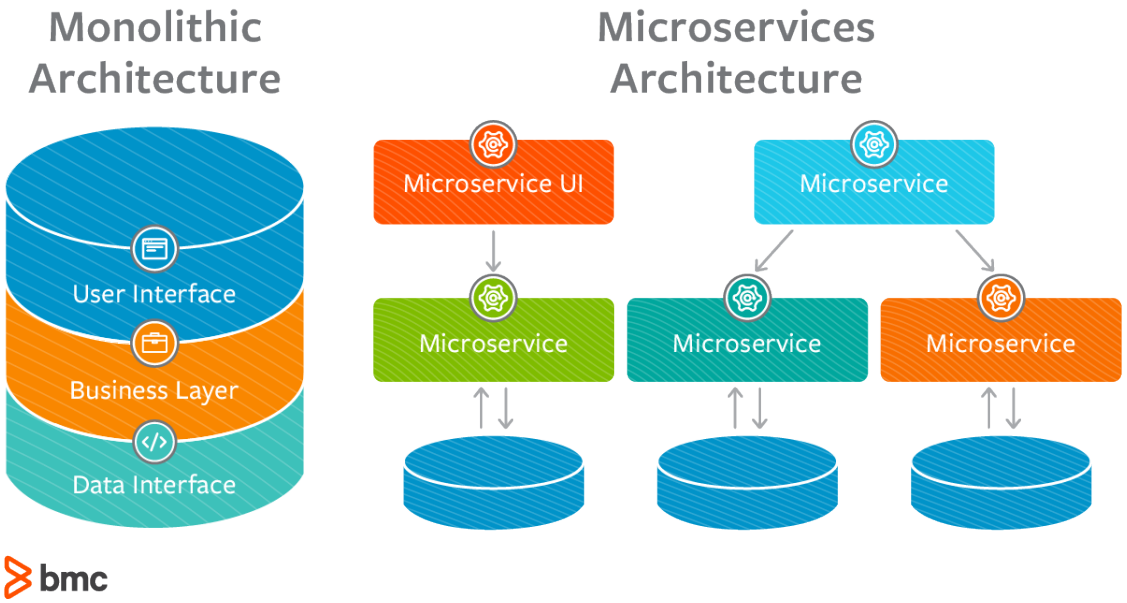

Figura 3.3: Diferenţa dintre arhitecturi

Chiar dacă mulţi dezvoltatori sunt reţinuţi în vderea unei schimbări, sau ezită să încerce o abordare diferită, beneficiile microserviciilor sunt mult mai importante şi mai semnificative decât dezavantajele, în cazul multor apliicaţii[5]. Arhitecturile modulare reduc riscul schimbarilor nedorite sau neanticipate dintr-o componentă în urma modificării altei componente. În concluzie, microserviciile sunt o parte miscării din industria IT care permite o colaborăre mai simplă între echipe. Microserviciile nu sunt doar o tehnologie folosită în prezent, ele sunt un o cultură despre procesul de dezvoltare software.

3.3 Microservicii în Cloud

Resursele în *Cloud* sunt disponiibile şi puse la dispoziţia atunci când sunt necesare. Comparativ cu o infrastructură clasică, nu există o limită practică a acestora. Diferite medii de dezvoltare şi versiuni de servicii pot co-exista îin mod temporar sau permanent. Programatorul nu mai este nevoit să ghească sau să calculeze cerinţele şi capacitatea de

consum a sistemului. La cerere resursele pot fi scalate sau diminuate fără intervenţie în partea fizică a sistemului.

Faptul că plăteşti doar ceea ce utilizezi reduce considerabil preţul experimentării, dar şi creşte posibilităţile de experimentare. Noi funcţiionalităţi pot fi integrate, pot fi oprite şi restartate cu noi parametri în cazul unui eşec. Datorită cloudului se pot efectua numeroase experimente fără riscuri, acets lucru constituind cheia de succes a inovaţiei. Acest fapt se potriveşte ideal cu conceptul de *microservicii*, oferind posibilitatea de a atinge un nivel înalt de agilitate.

Programabilitatea cloud-ului permite automatizarea proceselor de dezvoltare şi livrare. Integrarea contnuă este o parte a ciclului de viaţă a servicilor în cloud. Livrarea continuă, pe de altă parte, introduce provoci noi a complexităţi de administrarea a multiplelor servicii în paralel.

Gândirea, perfecţionarea continuă, livrarea continuă, managementul, monitorizarea şi întreţinerea API-urilor este o responsabilitate complexă şi consumă extrem de mult timp. Sistemele Cloud oferă suport pentru acestea şi uşurează viaţa dezvoltatorilor.

Arhitectura bazată pe microservicii este o abordare distribuită, dezvoltată pentru a rezolva limitările arhitecturii monolitice clasice. Microserviciile facilitează scalarea aplicaţiilor sau a anumitor module ale aplicaţiei. Totuşi sunt o provocare din punctul de vedere a complexităţii arhitecturale şi a operării sistemului. Serviciile cloyd contribuie la reducerea acestei complexităţi prin oferirea unor funcţionalităţi preimplementate de management al serviciilor.

3.4 Securitatea in sistemele informatice

Securitatea aplicaţiilor cuprinde măsurile luate pentru a asigura şi a îmbunătăţi securitatea sistemului. Deseori aceasta este asigurată prin găsirea, înlăturarea şi prevenirea vunlnerabilităţilor de securitate. Sunt utiilizate mai multe tehnici pentru a obţine acest lucru, acestea pot fi aplicate la diferite etape ale ciclului de dezvolatare, cum ar fi: *design, dezvoltare, livrare, perfecţionare sau mentenanţă.*

Sunt utilizate diferite abordări pentru a găsi diferite subseturi ale vulnerabilităţilor de securitate, acestea au un impact diferit prin cost, timp, efort şi procentul de vulnerabiliităţi ce pot fi detectate. Tehnicile esenţiale sunt:

- *Whitebox* - se referă la analiza securităţii sau a codului. Această abordare poate fi adoptată doar de un inginer care înţelege deplin codul, şi poate observa problemele prin analiza manuală şi vizuală a codului sursă.

- *Blackbox* - reprezintă auditul în securitate. Operaţia poate fi efectuată de un specialist în securitate, utilizând doar executabilul, fără necesitatea de analiza codul sursă. Inginerul se concentrează pe încercările de a găsi cazuri netratate care pot duce la compomiterea sistemului.

- *Revizuirea design-ului* - înainte de scrierea codului se analizează vulnerabilităţile arhitecturale şi ale dependenţelor externe ale sistemului.

- *Utilizarea tool-urilor* - în prezent există un număr mare de tooluri automate care pot fi utilizate pentru detectarea problemelor de securitate, însă acestea au un număr mai mare de fals pozitive decât în cazurile de testare manuală.

- *Platforme de vulnerabilităţi coordonate* - sunt aplicaţii care oferă recompense hackerilor experimentaţi pentru găsirea de probleme de securitate.

Utilizarea acestor tehnici în mod adecvat îmbunătăţeşte calitatea sistemului prin înlăturarea vulnerabilităţilor. Acest proces este în totalitate responsabilitatea echipei de dezvoltare.

În timp ce stocarea în sistemele cloud este convenabilă şi oferă posibilitatea de a accesa datele indiferent de locaţie şi oră, de pe aproximativ orice dispozitiv cu conexiune la internet, securitatea sistemelor de sticare este o problemă prioritară a organizaţiilor IT şi a departamentelor de securitate. Beneficiul principal al adoptării stocării în cloud este asigurarea securităţii si integriităţii datelor cu caractez senzitiv.

Dezvoltatorilor sistemelor de cloud le aparţine în totalitate responsabilitatea pentur securitatea aplicaţiilor lor. Aceştia implementează în sistemele lor toate funcţionalităţile esenţiale pentru securitate, acestea fiind: *autentificarea, autorizarea, controlul accesului şi criptarea*. De aici în colo, fiecare companie are responsabilitatea de a adăuga noi nivele de protecţie pentru date şi de a restricţiona cât mai mult accesul la datele sensibile.

3.4.1 Ameninţări

Administratorii de sistem şi dezvolatatorii de servicii software sunt mereu la straja securităţii aplicaţiilor. Însă sunt numeroase probleme care par netriviale, însă pot compromite datele apliicaţiei. Principalele amenintări pentru securitatea aplicaţiilor sunt:

- *Utilizatorii* - utilizatorii aplicaţiilor noastre sunt cea mai mare ameninţare pentru propria lor integritate şi pentru datele lor. Deseori aceştia nu realizează cât de important e să accesezi doar partea sigură a internetului. Nerespectarea unor reguli de bază a navigării pe internet poate compromite parole, chei de acces, chei de criptare, orice efort din partea dezvoltatorilor de a păstra securitatea va eşua în acest caz.

- *Greşeli elementare de structurare sau scriere a codului* - în ciuda multiplelor avertismente şi a anilor de educaţie încă există cod cu greşeli elementare de securitate. Problemele triviale sunt: *SQL-injection* şi*Cross-site scripting*.O recomandare în acestă direcţie este utilizarea unor librării specializate pentru SQL şi pentru randare a datelor provenite de la utilizatori, acestea tratează aceste cazuri şi aplicaţia nu poate fi atacată în acest mod.

- *Utilizarea unor librării învechite* - este recomandat să se utilizeze cele mai noi versiuni ale unor librării şi aplicaţii, aceste nu conţin de obicei erorile şi problemele de securitate cunoscute.

- *Setarea unor permisiuni de acces greşite* - prin setarea permisiunilor de acces greşite, utilizatorul poate obţine prea multă libertate şi control asupra aplicaţiei. Este recomandat ca permisiunile să fie la nivelul minim necesar pentru utilizarea aplcaţiei conform modelului de cazuri de utilizare.

- *Hackerii* - persoanele care doresc să obţină profit sau date preţioase vor încerca mereu să strice aplicaţia, problema nu poate fi înlăturat omplet, dar procesul poate fi făcut mai complex prin îndeplinirea unor norme de securitate mai avansate.

- *Lipsa obfuscării sau criptării datelor* - datele stocate îin formă citibilă crează o facilitate pentru hackerii, aceştia nu mai au nevoie de muncă suplimentară pentru obţinerea datelor valoroase odată ce au obţinut control asupra siistemului.

Măsurile esenţiale pentru securitate vor fi discutate în secţiiunile ce urmează.

3.4.2 Criptografia

Criptografia - este utilizată pentru ascunderea mesajelor. Există numeroase metode de criptare a datelor începând cu Cifrul lui Caesar, una dintre cele mai primitive metode de criptare exiistente, terminând cu AES şi RSA, acre sunt metodele de criptare standardizate, considerate aproape invincibile în momentul de faţă. Totuşi criptografia nu este o soluţie general ăpentru securitate, aceasta este privită mai mult ca un *tool*. Adversarul principal al criptografiei este - criptanaliza, ştinţa destinată descifrării mesajelor criptate prin analiza datelor de intrare şi ieşire ale unui algoritm.

3.4.3 Compresia

3.4.4 Steganografia

3.5 Sisteme similare

Acest capitol reprezintă clasificarea si analiza sistemelor similare existente, bazată pe etapa de cercetare a proiectului. Sistemele au scop şi functionalităţi similare cu proiectul propus. Sistemele alese pentru comparaţie sunt:

- CloudMe
- Dropbox
- CrashPlan

- ICloud
- Google Drive
- OneDrive
- pCloud
- sync.com

Dropbox, Google Drive, ICloud si OneDrive au fost incluse în acest studiu deoarece sunt în top 10 cele mai populare servicii de cloud 2019 [6]. Acestea reprezintă un model pentru cum este văzut un cloud storage modern: simplu de configurat, simplu de utilizat şi disponibil la un preţ avantajos. pCloud si sync.com sunt în topul sistemelor cu cea mai înaltă recuritate de pe piaţă. pCloud este categorizat ca un sistem infraudabil şi nu a avut nici o expunere a datelor utilizatorilor. Însă aceste sisteme vin si cu preţuri de 3-4 ori mai mari decât sistemele clasice.

3.5.1 Metodologia de analiză

În ultimul deceniu tot mai mulţi utilizatori, atât business cât şi individuali, se basează pe stocarea fişierelor în Cloud. Cele mai importante criterii pe care se bazează utilizatorii sunt: securitatea, simplitatea de utilizare a sistemului, disponibilitatea si preţul de utilizare. Analiza sistemelor individuale de cloud a fost efectuată în modul următor:

Sunt sumarizate preţurile de utilizare a sistemului şi a diferitor opţiuni. Sunt analizate detaliile capabilităţilor tehnice şi organizaţionale ale părţii client şi server a sistemului. Informaţiile colectate sunt bazate pe secţiunile *Terms of Service* şi *Privacy Policy* ale documentaţiilor oficiale ale sistemelor[7]. Rezultatele analizei comparative au ca scop determinarea cerinţelor pricipale ale unui sistem de stocare cloud şi comparaţia sistemului elaborat cu cele existente. În secţiunile ce urmează se vor analiza următoarele funcţionalităţi:

Tabelul 3.1: Criterii evaluare sisteme similare

Copy	Backup	Sync	Sharing	Client-side Encryption	Server-side encryption	Compression	Watermarking

Pentru fiecare categorie din tabelul 3.1 se va acorda un punctaj conform următoarelor reguli:

✓✓ este echivalent pentru *foarte bine*, toate cerinţele obligatorii pentru funcţionalitatea respectivă au fost îndeplinite şi câteva dintre cele opţionale.

✓ este echivalent pentru *bine*, adică toate cerinţele obligatorii pentru funcţionalitatea respectivă au fost îndeplinite.

± este simbolul pentru *bine cu câteva vulnerabilităţi*, nu toate cerinţele esenţiale au fost îndeplinite.

✗ este echivalent cu *slab*, cel puţin o cerinţă obligatorie nu a fost îndeplinită.

✗✗ este echivalentul pentru *foarte slab*, adică mai multe dintre cerinţele obligatorii nu sunt indeplinite in funcţionalitatea respectivă sau funcţionalitatea lipseşte.

3.5.2 CloudMe

CloudMe[8] este un sistem de cloud standard, care vine cu un serviciu de sincronizare şi backup a fişierelor, după o analiză detaliata ne putem da seama că acest sistem a fost inspirat din arhicunoscutul **Dropbox** care este analizat în secţiunea 3.5.3.

În tabelul 3.2 sunt prezentate funcţionalităţile sistemului şi o notă a fiecărei funcţionalităţi în concordanţă cu evaluările primite pe pagina oficială a sistemului, precum şi a vulnerabilităţilor depistate în ultima perioadă.

Tabelul 3.2: CloudMe Funcţionalităţi

Copy	Backup	Sync	Sharing	Client-side Encryption	Server-side encryption	Compression	Watermarking
Da ✓	Da ✓	Da ✗✗	Da ±	Da ✓✓	Nu ✗✗	Nu ✗✗	Nu ✗✗

Tabelul 3.3 prezintă disponibilitatea **CloudMe** pe diferite platforme şi pretul acestuia.

Tabelul 3.3: CloudMe Platforme disponibile şi preţ

Web Client	Desktop client	Mobile Client	500GB Plan Price
Da	Da	Da	10€

Un fapt bun despre **CloudMe** este că acesta pune la dispoziţia utilizatorului un spaţiu de stocare gratuit de 3GB, iar pentru volume de date mai mari oferă opţiuni la preţul mediu al pieţei. Serviciul oferă funcţionalităţile de bază, însă nimic desebit în materie de securitate.

Interfaţa web a **CloudMe** este foarte simplă si clară, este evident cum sa încarci un fisier sau cum sa îl schimbi în alt folder.

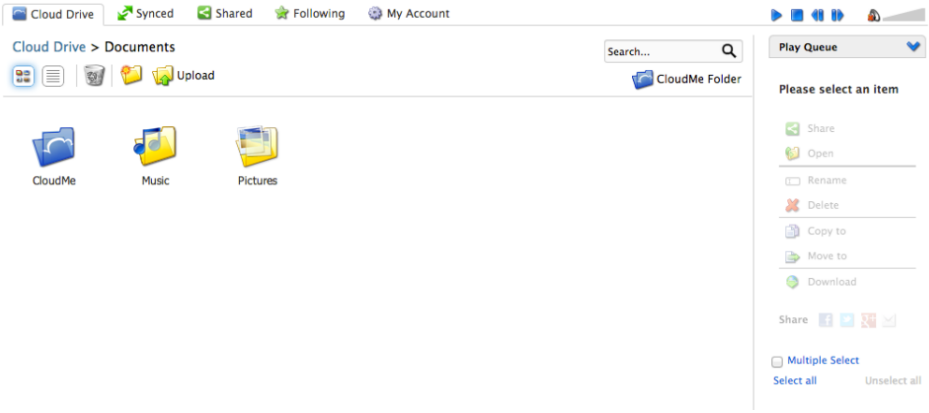

Figura 3.4: CloudMe - interfaţa web

CloudMe oferă sincronizarea aplicaţiilor pentru Windows, Mac, Linux, iOS şi Android. Pentru ca sincronizarea sa poata avea loc utilizatorul primeşte un aşa numit "director albastru" oferă utilizatorului singronizare în timp real. De asemenea, **CloudMe** oferă opţiunea de a alege timpul la care se va întâmpla sincronizarea. Pentru a distribui fişiere sunt disponibile mai multe opţiuni, începând cu distribuire clasică, care permite utilizatorului să ofere acces la fişierele lui prin distribuirea unui link, şi ajungând la metode mult mai colaborative care permit altor utilizatori, cărora li se oferă access, să modifice fişierele din cloud-ul altui utilizator sau să încarce fişiere noi.

Figura 3.5: CloudMe - planuri de preţ

CloudMe arhivează versiunile precedente ale fişierelor prin funcţionalitatea coşului de gunoi, care păstrează pentru 60 de zile toate fişierele care au fost şterse.

În privinţa securităţii **CloudMe** nu este o opţiune prea bună deoarece nu oferă criptarea datelor, acestea fiind vulnerabile pentru atacuri. Este posibil să încarci şi să descarci fişiere criptate, însă criptarea şi decriptarea acestora ramăne la latitudinea utilizatorului.

CloudMe este o aplicaţie foarte uşor de utilizat şi are o interfaţă foarte intuitivă. Acesta dispune de funcţionalităţile de bază ale unui sistem de stocare în cloud, însă nu este

potrivit pentru stocarea fisierelor cu conţinut de date senzitiv din cauza lipsei criptării, de asemnea atunci când fişierele sunt distribuite după un link este foarte greu să determini cine a făcut public un fişier cu caracter privat deoarece link-ul de acces poate fi uşor furat.

Un alt defect al acestui sistem este funcţionalitatea de *Sync*, motivul pentru care i-am oferit punctaj minim este că în ultimul an a avut multiple vulnerabilităţi, conform *CVE-2018-6892* [1] atacatorii se puteau conecta la clientul de "CloudMe Sync" prin portul 8888 şi trimiterea unor date maliţioase puteau cauza "buffer overflow", acest lucru le oferea control asupra execuţiei şi posibilitatea de a executa cod maliţios.

3.5.3 Dropbox

Dropbox a fost lansat in 2007 şi este definit ca unul dintre cele mai bune servicii de cloud pentru uz general[9]. Sistemul are peste 500 de milioane de utilizatori în toată lumea, fiind unul dintre cele mai competitive servicii de pe piaţă.

În tabelul 3.4 este prezentată o evaluare a sistemului pe baza unei evaluări personale, dar şi pe baza evaluărilor oferite de *CloudWards*[9] şi *On the Security of Cloud Storage Services*[7].

Tabelul 3.4: Dropbox Funcţionalităţi

Copy	Backup	Sync	Sharing	Client-side Encryption	Server-side encryption	Compression	Watermarking
Da ✓	Da ✓	Da ✓	Da ✓✓	Nu ✗✗	Da ±	Nu ✗✗	Nu ✗✗

În tabelul 3.5 sunt prezentate opţiunile de clienţi disponibili pentru **Dropbox** şi oferta de preţ.

Tabelul 3.5: Dropbox Platforme disponibile şi preţ

Web Client	Desktop client	Mobile Client	500GB Plan Price
Da	Da	Da	18€

Dropbox este uşor de utilizat atăt utilizând aplicaţia web, cât şi cea mobile sau desktop. În figura 3.6 este prezentată interfaţa web a sistemului.

[1] https://nvd.nist.gov/vuln/detail/CVE-2018-6892

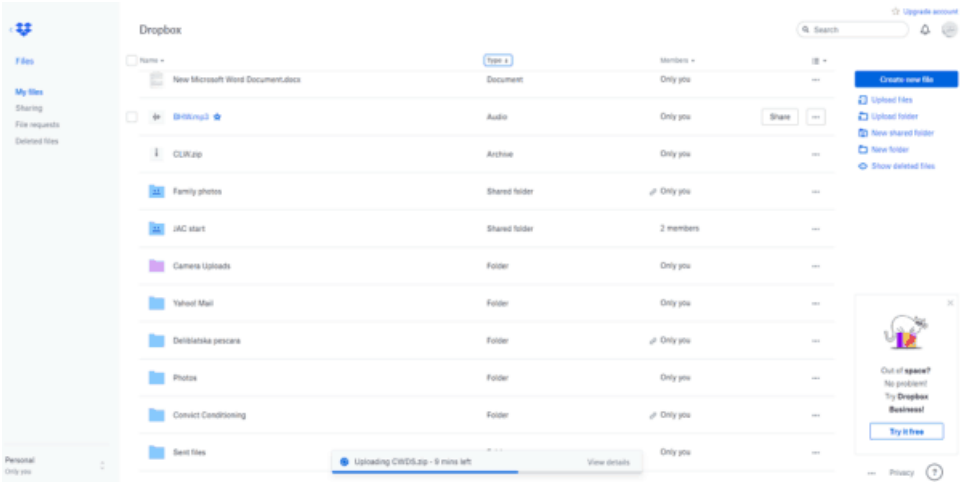

Figura 3.6: Dropbox - interfaţa web

Dropbox oferă un serviciu de sincronizare ”rapid şi inteligent”, cu diverse posibilităţi de customizare, însă această opţiune poate fi aplicată doar pe anumite directoare, lucru care restrânge libertatea utilizatorului.

Funcţionalitatea de partajare a fişierelor este în topul celor disponibile pe piaţă deoarece oferă partajare atât de fişiere cât şi de directoare, cu parolă sau fără, cu customizare de permisiune şi termen de expirare. Se pot trimite link-uri pe mail sau prin copierea directă, link-urile pot fi şterse şi fişierul nu mai este accesibil prin acel link. Această funcţionalitate este disponibilă atât din orice aplicaţie **Dropbox**.

O funcţionalitate a cărui autor este **Dropbox** este deduplicarea la nivel de bloc, prin împărţirea fişierului la încărcare în multiple porţiuni de dimensiune fixă şi prin scanarea dacă porţiunea există, acest lucru oferă o viteză de încărcare mai mare, dar prezintă un risc deoarece un atacator poate determina conţinutul unui fişier de anumit format prin încărcări repetate de conţinut diferit a anumitor porţiuni, de exemplu un fişier cu analize medicale.

La nivel de securitate **Dropbox** nu este cea mai bună soluţie, având un trecut destul de bogat în atacuri, suferind numeroase furturi de date. **Dropbox** salvează fişierele în format criptat, însă numeroase metadate care includ şi porţiuni de text sunt salvate in format text. Acest lucru nu este benefic pentru utilizatori deoarece datele lor pot fi uşor compromise.

Dropbox a avut o evoluţie specaculoasă în ultimii 6 ani, adaugând numeroase noi funcţionalităţi şi devenind mult mai uşor de utilizat. Însă **Dropbox** rămâne o soluţie pentru utilizatorii care nu au date senzitive stocate în acest sistem. Există mai multe variante de compromitere a datelor, una dintre acestea este cauzată de implementarea FTS, prin pastrarea metadatelor pentru căutare in format necriptat. Alta vulnerabilitate este cauzată de deduplicarea la nivel de bloc ce se execută la nivel de întreg sistem în loc de nivel fişiere utilizator.

O vulnerabilitate de securitate a fost descoperită în 2018 *CVE-2018-12271*[2] atunci când un atacator se putea loga pe un cont **Dropbox** cu orice amprentă arbitrară şi avea access la toate fişierele utilizatorului, un nivel suplimentar de securitate prin criptare cu cheie provenită de la utilizator ar fi prevenit acest lucru.

3.5.4 CrashPlan

CrashPlan este un sistem de stocare a fişierelor şi backup care a apărut pe piaţă în 2007. În prezent, sistemul a devenit unul foarte popular, zilnic acesta procesează peste 100 de miliarde de fişiere.

În tabelul 3.6 sunt prezentate funcţionalităţile sistemului şi o evaluare efectivă a acestor capabilităţi.

Tabelul 3.6: CrashPlan Funcţionalităţi

Copy	Backup	Sync	Sharing	Client-side Encryption	Server-side encryption	Compression	Watermarking
Da ✓	Da ✓	Nu ✗✗	Nu ✗✗	Da ✓✓	Da ✓✓	Nu ✗✗	Nu ✗✗

Din evaluarea de mai sus se observă că sistemul nu oferă partajare de fişiere, în realitate această funcţionalitate a fost înlăturată recent, cauza fiind riscul de compromitere a datelor. De asemenea partajarea fişierelor şi menţinerea unui nivel de securitate crescut implică un efort considerabil, de aceea, pentru moment, serviciul a fost deactivat.

În tabelul 3.7 sunt prezentate platformele pe care este disponibil sistemul, se poate observa că preţul este mai mic comparabil cu adversarii analizaţi în secţiunile precedente, acest fapt se poate datora lipsei unor anumite funcţionalităţi.

Tabelul 3.7: CrashPlan Platforme disponibile şi preţuri

Web Client	Desktop client	Mobile Client	500GB Plan Price
Da	Da	No	10€

CrashPlan a fost caracterizat ca unul dintre cele mai bune sisteme pentru backup[10] datorită factorului că nu are dimensiune maximă a fişierelor pentru backup. Crashplan nu necesită implicare din partea utilizatorului pentru a executa copierea regulată a fişierelor. De asemenea, CrashPlan permite executarea operaţiunii de backup în acelaşi cont de utilizator a până la 10 calculatoare.

CrashPlan oferă mai multe nivele de securitate pentru fişiere, toate datele sunt criptate de la client până la server. Criptarea se execută utilizând o cheie de 448 biţi pentru utilizatorii unui plan plătit, pentru utilizatorii opţiunii gratuite se utilizează o

[2]https://www.cvedetails.com/cve/CVE-2018-12271/

cheie de 128 biţi. Cheile de criptare sunt generate utilizănd un sistem eficient de numere aliatoare. De asemena există opţiunea de a selecta o cheie privată de criptare, acesta nu va fi salvată niciodata sub formă de text şi nu va fi accesibilă nimănui.

Cel mai important lucru care face **CrashPlan** un sistem extrem de bun este performanţa şi eficienţa criptării care oferă o securitate excepţională a datelor, acestea nu pot fi decriptate fără cunoştinţa şi acordul utilizatorului. Datele utilizatorilor au fost compromise o singură dată, din cauza unui vulnerabilităţi ce permitea executarea codului la distanţă, acest lucru a devenit posibil din cauza unei vulnerabilităţi din clasa Java *DateRMI*, descrierea vulnerabilităţii poate fi găsită în *CVE-2017-9830* [3].

3.5.5 ICloud

ICloud este unul dintre cele mai populare şi utilizate sisteme de cloud conform CloudWards [11], acest fapt nu este datorat doar poziţiei monopolistice pe care o ocupă pe piaţă, ci şi funcţionalităţilor pe care le oferă.

Sistemul **ICloud** este preinstalat pe toate dispozitivele Apple, acest lucru este deseori privit ca motivul pentru care este atât de popular. Totuşi, conform analizei funcţionalităţilor, care este prezentată în tabelul 3.8 se poate observa că funcţionalităţile acestuia se ridică la un nivel destul de înalt.

Tabelul 3.8: ICloud Funcţionalităţi

Copy	Backup	Sync	Sharing	Client-side Encryption	Server-side encryption	Compression	Watermarking
Da ✓	Da ✓	Da ✓✓	Da ±	Da ✓✓	Nu ✗✗	Nu ✗✗	Nu ✗✗

În tabelul 3.9 sunt prezentate platformele pe care este disponibil **ICloud**.

Tabelul 3.9: ICloud Platforme disponibile şi preţuri

Web Client	Desktop client	Mobile Client	500GB Plan Price
Da	Da	Da	5€

Însă acesta are şi câteva probleme legate de clientul Desktop, care are funcţionalităţi limitate, şi limitarea funcţionalităţii de partajare de fişiere.**ICloud** oferă funcţionalitatea de sincronizare cu Apple Photos. Iar funcţionalitatea de share poate fi accesată din orice fişier care se află într-un director sincronizat.

În 2014 a avut loc un furt de date de dimensiuni foarte mari, acesta a compromis reputaţia ICloud, însă atacul a fost făcut prin forţă brută şi ”pescuirea datelor” de la viitoarele victime. În realitate apple oferă câteva funcţionalităţi de securitate care îl fac

[3] https://www.cvedetails.com/cve/CVE-2017-9830/

un sistem cu nivel de securitate peste media de pe piaţă. Însă Apple nu este un sistem cu ”cunoştinţă zero”, acesta stochează cheile de criptare în acelaşi loc cu fişierele criptate, asta íl face extrem de vulnerabil în cazul unui atac.

Spre deosebire de alte sisteme, cum ar fi *Google*3.5.6, este bine cunoscut că *Apple* nu colaborează cu guvernul sau companiile publicitare şi nu va oferi informaţii private despre clienţii săi.

ICloud nu este o soluţie genrală pentru stocarea fişierelor, acesta este potrivit pentru utilizatorii care nu au nevoie să păstreze date extrem de sensibile din cauza posibilităţii de decriptare prin brute force. Acesta nu este potrivit nici pentru utilizatorii care doresc o viteză ridicată de încărcare şi decărcare a fişierelor, însă Apple continuă să se perfecţioneze şi să crească viteza operaţiilor de reţea.

De asemenea iCloud este extrem de vulnerabil pentru executare de cod la distanţă conform datelor oferite de CVEDetails.com, vulnerabilităţile recent descoperite sunt *CVE-2018-20506*[4] şi *CVE-2018-4464*[5], acest fapt este datorat popularităţii sistemului crae îl face o ţintă importantă pentru hackeri.

3.5.6 Google Drive

Cu aproximativ un miliard de utilizatori, **Google Drive** este cel mai popular serviciu de *cloud* de pe piaţă, această popularitate nu este datorată doar faptului că este preinstalat pe telefoanele Android, dar şi capabilităţilor de partajare şi vitezei înalte de decărcare şi încărcare a fişierelor.

În tabelul 3.10 sunt prezentate evaluări ale funcţionalităţilor sistemului.

Tabelul 3.10: Google Drive Funcţionalităţi

Copy	Backup	Sync	Sharing	Client-side Encryption	Server-side encryption	Compression	Watermarking
Da ✓	Da ✓✓	Da ✓	Da ✓	Da ✓✓	Da ✓	Da ±	Nu ✗✗

Deşi **Google Drive** oferă criptarea datelor, securitatea si intimitatea nu sunt punctele forte ale sistemului, acesta având antecedente de implicare în campaniile miltare de colectare a datelor şi spionării cetăţenilor. Compresia datelor, pe de altă parte, se referă la reducerea dimensiunii imaginilor, proces de compresie cu pierderi.

În tabelul 3.11 sunt prezentate opţiunile de client disponibile pentru **Google Drive**, de asemenea fiecare utilizator primeşte iniţial 10 GB de stocare gratuită.

[4]https://www.cvedetails.com/cve/CVE-2018-20506/
[5]https://www.cvedetails.com/cve/CVE-2018-4464/

Tabelul 3.11: Google Drive Platforme Disponibile şi Preţuri

Web Client	Desktop client	Mobile Client	500GB Plan Price
Da	Da	Da	15€

Google Drive este unul dintre cele mai bune sisteme pentru colaborare, ocupând locul 2, după Dropbox, prezentat în secţiunea 3.5.3. Punctul forte al colaborării oferite de Google Cloud este faptul că Office Suite este integrat în acest sistem.

Funcţionalitatea de sincronizare este extrem de performantă, dar nu oferă sincronizare la nivel de bloc de fisier. Partajarea de fişiere este una dintre cele mai folosite funcţionalităţi ale sistemului, însă acesta vine cu câteva vulnerabilităţi cauzate de lipsa criptării la partajare sau protejarea cu parolă. Partajarea este uşor de executat link-ul poate fi partajat prin email, Facebook, Twitter sau copiat şi salvat în destinaţia dorită.

Datele sunt criptate utilizând AES-128 atunci când sunt salvate pe disc şi prin TSL atunci când sunt transportate între client şi server. Oricum sistemul nu oferă "zero-knowledge" şi oricine deţine are control asupra sistemului poate să citească datele (exemplu: programatorii sistemului). Totuşi pentru o protecţie mai bună Google oferă posibilitatea de autentificare în 2 paşi.

Google Drive vine cu foarte multe aplicaţii integrate şi posibiltăţi de colaborare integrate în sistem. Autentificare în 2 paşi, criptare a datelor şi viteză ridicată de încărcare şi descărcare, dar lipsa posibilităţii de criptare privată şi vulnerabilităţile cauzate de lipsa unei criptări destul de sigure pentru funcţionalitatea de partajare de fişiere fac **Google Cloud** sa nu fie cea mai bună soluţie pentru securitatea şi integritatea datelor.

3.5.7 OneDrive

OneDrive este un sistem de stocare în cloud dezvoltat de compania MIcrosoft, are un trecut destul de ambiguu în privinţa securităţii datelor, deşi şi-a perfecţionat securitatea, nu poate fi încadrat în topul celor mai sigure sisteme de securitatea[**?**], însă tinde spre acesta.

Analiza funcţionalităţilor sistemului este prezentată în tabelul 3.12.

Tabelul 3.12: OneDrive Funcţionalităţi

Copy	Backup	Sync	Sharing	Client-side Encryption	Server-side encryption	Compression	Watermarking
Da ✓	Da ✓✓	Da ✓	Da ✓	Nu ✗✗	Da ✗✗	Nu ✗✗	Nu ✗✗

Funcţionalităţile nu au primit o notă maximă din cauza ambiguităţii de utilizare a sistemului, în dependenţă de tipul fişierului selectat, meniurile arată diferit, uneori poate fi o problemă pentru utilizatori. De asemenea, sistemul nu este unul cu cunoştinţe zero,

atacatorul poate obţine datele unui utilizator, odată ce s-a infiltrat în sistem, deoarele toate datele necesare pentru decriptare sunt deja prezente în sistem.

În tabelul 3.13 sunt prezentate datele cu privire la disponibilitatea sistemului pe diferite platforme, dar şi preţul unui abonament cu spaţiu de stocare de 500GB.

Tabelul 3.13: OneDrive Platforme Disponibile şi Preţuri

Web Client	Desktop client	Mobile Client	500GB Plan Price
Da	Da	Da	15€

OneDrive urmează nişte principii definite de Dropbox cu privire la standardele de sincronizare a fişierelor. OneDrive obişnuia să aibă probleme serioase cu privire la securitate, neavând nici criptare la nivelul nivelului de stocare. În prezent, ânsă sistemul oferă criptare la transmitere şi la stocare. Criptarea nivelului de stocare include 2 componente esenţiale: *BitLocker*, criptare la nivel de disc, şi un sistem de criptare per fişier a conţinutului. Fiecare fişier este securizat priin utilizarea unei chei AES unice de 256 de biti, de asemenea se foloseşte protocolul TLS pentru a preveni atacurile de tipul *man-in-the-midle* Dar minusul acestui sistem este că cheile sunt stocate pe acelaşi sistem, orice angajat poate eventual să citească datele utilizatorilor sau să le ofere companiilor de e-publicitate sau unor servicii secrete.

În concluzie, **OneDrive** este un sistem care a evoluat considerabil în ultima perioadă, însă are neajunsuri considerabile pe partea de securitate. De asemenea sistemul nu oferă funcţionalitate de comopresie, ceea ce creşte costulm de stocare a datelor, datele pot fi compresate de utilizator în prealabil, iar după descărcare acestea pot fi decompresate, operaţia însă este anevoioasă şi prezintă o problemă în cazul partajării fişierelor. Sistemul nu este potrivit pentru stocarea unor date sensibile şi este recomandat doar pentru stocarea unor date netriviale.

3.5.8 pCloud

pCloud a fost fondat în 2013 şi în doar 3 ani a ajuns la peste 3 milioane de utilizatori, competitorii cei mai importanţi sunt Dropbox şi Copy. Chiar dacă este nou pe piaţă, spre deosebire de competitorii săi mari, acesta oferă funcţionalităţi de top şi este inclus în topul *Most Secure Cloud Storage 2019: Safety First*[**?**].

pCloud oferă funcţionalităţile de sincronizare, backup (se opoate aplica şi pe datele de pe Instagram sau Facebook), colaborare şi criptare avansată a datelor. O evaluare a acestor capabilităţi este oferită în tabelul 3.14.

Tabelul 3.14: Funcţionalităţi

Copy	Backup	Sync	Sharing	Client-side Encryption	Server-side encryption	Compression	Watermarking
Da ✓	Nu ✗✗	Da ✓	Da ✓	Da ✓✓	Da ✓✓	Nu ✗✗	Nu ✗✗

Sistemul este prezent pe toate tipurile de platforme, tabelul 3.15 şi are un preţ destul de ridicat, însă rezonabil pentru un sistem securizat de stocare.

Tabelul 3.15: Sisteme de operare

Web Client	Desktop client	Mobile Client	500GB Plan Price
Da	Da	Da	40€

Securitatea este unul dintre punctele forte ale sistemului, dezvoltatorii încearcă să ofere servicii la cel mai înalt nivel şi sa ofere funcţionalităţi care nu sunt prezente la competitori. Sistemul foloseşte key de 256 biti şi TSL pentru transmisia de date. Securitatea însă vine şi cu un preţ, fişierele nu pot fi modificate în sistem, deoarece operaţiile de criptare şi decriptare ar trebui efectuate repetat, provocând costuri de prelucrare ridicate. De asemenea, sediul central al companiei se află în Elveţia, unde sunt cele mai stricte reguli de protecţie a datelor, astfel utilizatorul poate fi liniştit cu privire la integritatea datelor sale.

Datele încărcate sunt repartizate în funcţie de tipul de date: imagini, documente, muzica şi video.

Un dezavantaj al sistemului este că criptarea este o funcţionalitate care nu este oferită în pachetul de bază şi are un preţ mai ridicat, comparativ cu alte sisteme care oferă criptarea ca serviciu gratuit.

3.5.9 sync.com

Sync.com a fost fondat în 2011 de către Suhan Shan, Thoman Savundra, şi Darius Antia. Acesta a devenit deja un competitor serrios pentru Google Drive si Dropbox. Caracteristicile pentru care a ajuns atât de popular sunt analizate în tabelul 3.16.

Tabelul 3.16: Funcţionalităţi

Copy	Backup	Sync	Sharing	Client-side Encryption	Server-side encryption	Compression	Watermarking
Da ✓	Da ✓✓	Da ✓	Da ✓	Da ✓✓	Da ✓✓	Nu ✗✗	Nu ✗✗

Din păcate sistemul nu oferă compresia sau deduplicarea datelor, sistemul stocând volumul real de date pe care îl primeşte de la utilizator, plus câteva metadate create de algoritmii de compresie.

Tabelul 3.17: Sisteme de operare

Web Client	Desktop client	Mobile Client	500GB Plan Price
Da	Da	Da	15€

Criptarea datelor face ca sistemul să nu mai poată interpreta datele stocate, astfel sistemul nu oferă posibilitatea de deschidere şi vizualizare a fişierelor, doar decărcarea şi manipularea lor ulterioară cu alte aplicaţii. Însă există şi opţiunea de a stoca fişierele fără criptare. Caracteristicile cheie ale sync.com sunt :

- Cunostinţe zero despre date
- Criptare privată
- Uşor de utilizat
- Syncronizare
- Partajare de fişiere cu control asupra operaţiei

Sync este unul dintre sistemele care garantează securitate la nivelul cel mai înalt, acest sistem foloseşte pentru criptare RSA cu chei între 512 si 2048 biti. Sync nu păstrează cheile de criptare în sistem, daca utilizatorul îşi uită parola atunci datele lui nu vor putea fi recuperate niciodată. De asemenea se poate activa şi opţiunea de autentificare în doi paşi, pentru a oferi o protecţie şi mai bună.

Sistemul oferă funcţionalitatea de sincronizare, însă nu orice director poate fi selectat pentru efectuarea operaţieidin cauza faptului că se iau în considerare particularităţile fiecărei aplicatii. Sync ofera funcţionalitatea de partajare de fişiere care poate fi configurată adăugând parolă sau timp de expirare.

Nu putem nega că sync.com oferă o funcţionalitate de criptare complexă şi eficientă care oferă o securitate avansată, dar, partea negativă a acestui lucru este că atunci când dorim să vizualizăm un fişier sau săîl descărcăm, operaţia ar putea dura între câteva secunde şi cateva minute din cauza complexităţii adăugate de criptare.

3.5.10 Concluzii şi plasarea sistemului

Volumul de date stocate în cloud a crescut cu un factor de 40 în ultimii 10 ani, creşterea este constantă. Evoluţia tehnologică aduce un preţ mai mic pentru componentele hardware de stocare, dar şi cantităţi mai mari de date ceea ce împiedică scăderea preţului

serviciilor. De asemnea, evoluţia tehnologică aduce un impact negativ şi asupra securităţii, un atac de forţă brută poate fi executat mult mai uşor pe un sistem mai performant.

După studiul efectuat, am determinat că nici un sistem nu oferă functionalitatea de compresie de fişiere, cu toate că unele sisteme oferă deduplicare, aceasta vine cu un impact asupra securităţii utilizatorilor. Sistemul propus oferă reducerea volumului de date prin algoritmi de compresie şi decompresie fără pierderi, spre deosebire de Google care efectuează compresia imaginilor în versiunea gratuită prin reducerea dimensiunii, se pierde calitatea imaginii şi este o experientă neplăcută pentru utilizatori. Analiza comparativă este prezentată în tabelul 3.18.

Tabelul 3.18: Comparaţie sisteme similare

Nume	Copiere	Partajare	Criptare la partajare	Criptare pe disc	Algoritmi adaptivi	Compresie	Steganografie
CloudMe	✓	✓	✓	✗	✗	✗	✗
Dropbox	✓	✓	✗	✓	✗	✓	✗
CrashPlan	✓	✗	✗	✓	✗	✗	✗
ICloud	✓	✓	✗	✓	✗	✗	✗
GDrive	✓	✓	✗	✓	✗	✓	✗
OneDrive	✓	✓	✗	✗	✗	✗	✗
pCloud	✓	✓	✓	✓	✗	✗	✗
sync.com	✓	✓	✓	✓	✗	✗	✗
Bitstored	✓	✓	✓	✓	✓	✓	✓

Majoritatea sistemelor oferă criptarea datelor, unele au metode imposibil de spart, altele au metode mai simple şi puţin sigure. Sistemul propus criptează datele utilizatorului utilizând algoritmi auto-calibrabili care evoluează în funcţie de sistemul pe care ruleaza aplicaţia sau de trecerea timpului. De asemenea sistemul foloseşte cei mai noi si siguri algoritmi TwoFish şi PBFK2, care au fost modificaţi să folosească key mai sigure de o lungime de 256 şi 512 biţi.

Un aport nou pe care îl aduce sistemul şi nu a fost observat la niciuna dintre aplicaţiile studiate este semnătura pe fişiere, pentru a detecta furtul de date prin extragerea codului - tehnica folosită este steganografia.

Sistemul elaborat este mai mult un prototip, care poate fi dezvoltat, devenind un competitor pentru cele enumerate mai sus. Se pot adăuga funcţionalităţi suplimentare pe partea de partajare de fişiere.

Capitolul 4

Analiză şi Fundamentare Teoretică

4.1 Cazuri de utilizare

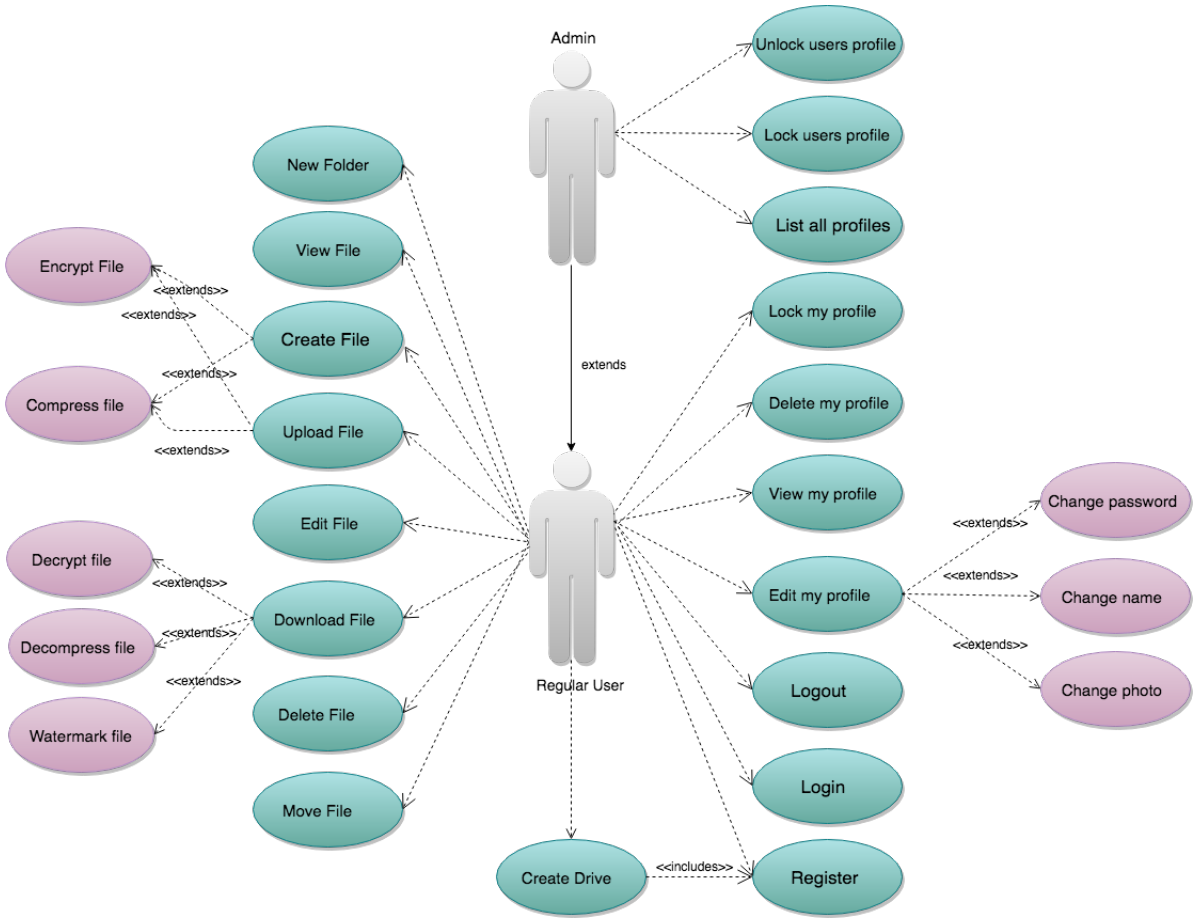

Figura 4.1: Use cases

4.1.1 Actori

4.1.2 Modele de cazuri de utilizare

4.2 Arhitectura canceptuală a sistemului

4.3 Tehnologii

4.3.1 Golang

Golang este un limbaj de programare care a apărut în 2007. Este un limbaj care care ofera analiză statică a codului: *gofmt* - formatarea statică a codului, *golint* - stilizarea codului, *godoc* - documentarae codului, acestea oferă o siguranţă asupra codului scris. În figura 4.2 este reprezentată emblema oficială a Golangului.

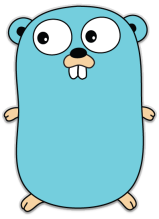

Figura 4.2: Golang

Go oferă un tool pentru testare integrat în limbaj, acesta a fost elaborat pentru simplitate şi oficienţă. Golang oferă un API foarte simplu care poate fi folosit pentru orice tip de teste.

4.3.2 gRPC

4.3.3 VueJS

4.3.4 MongoDB

4.3.5 Couchbase

4.3.6 HTML, CSS, Bootstrap

4.3.7 JSON Web Token(JWT)

4.3.8 Docker şi Kubernetes

4.3.9 Google Cloud

4.3.10 Git

Capitolul 5

Proiectare de Detaliu şi Implementare

Acest capitol prezintă deciziile şi paşii de implementare parcurşi în ciclul de dezvoltare al proiectului. Sistemul propus este format din 3 subsisteme: aplicaţia web; serverul, care este format din alte subsisteme, şi bazele de date. Capitolul va oferi o descriere succintă a tuturor componentelor, incluzând şabloanele arhitecturale, şabloanele de design şi algoritmii folositi în dezvoltarea proiectului.

5.1 Arhitectura serverului

5.1.1 Descriere generală

Pentru dezvoltarea aplicaţiei de server, principala tehnologie folosită a fost Golang, pentru unul dintre module s-a folosit Python. Pentru structura proiectului am ales şablonul arhitectural bazat pe microservicii, şablonul şi avantajele au fost descrise în secţiunea 3.2. Secviciile din care este compus serverul sunt:

1. Authentication service (descris în 5.1.3)
2. Compression service (descris în 5.1.6)
3. Crypto service (descris în 5.1.4)
4. File service (descris în 5.1.7)
5. Logging service (descris în **??**)
6. Watermarking service (descris în 5.1.5)
7. User service (descris în 5.1.8)

Arhitectura sistemului este prezentată în figura 5.1

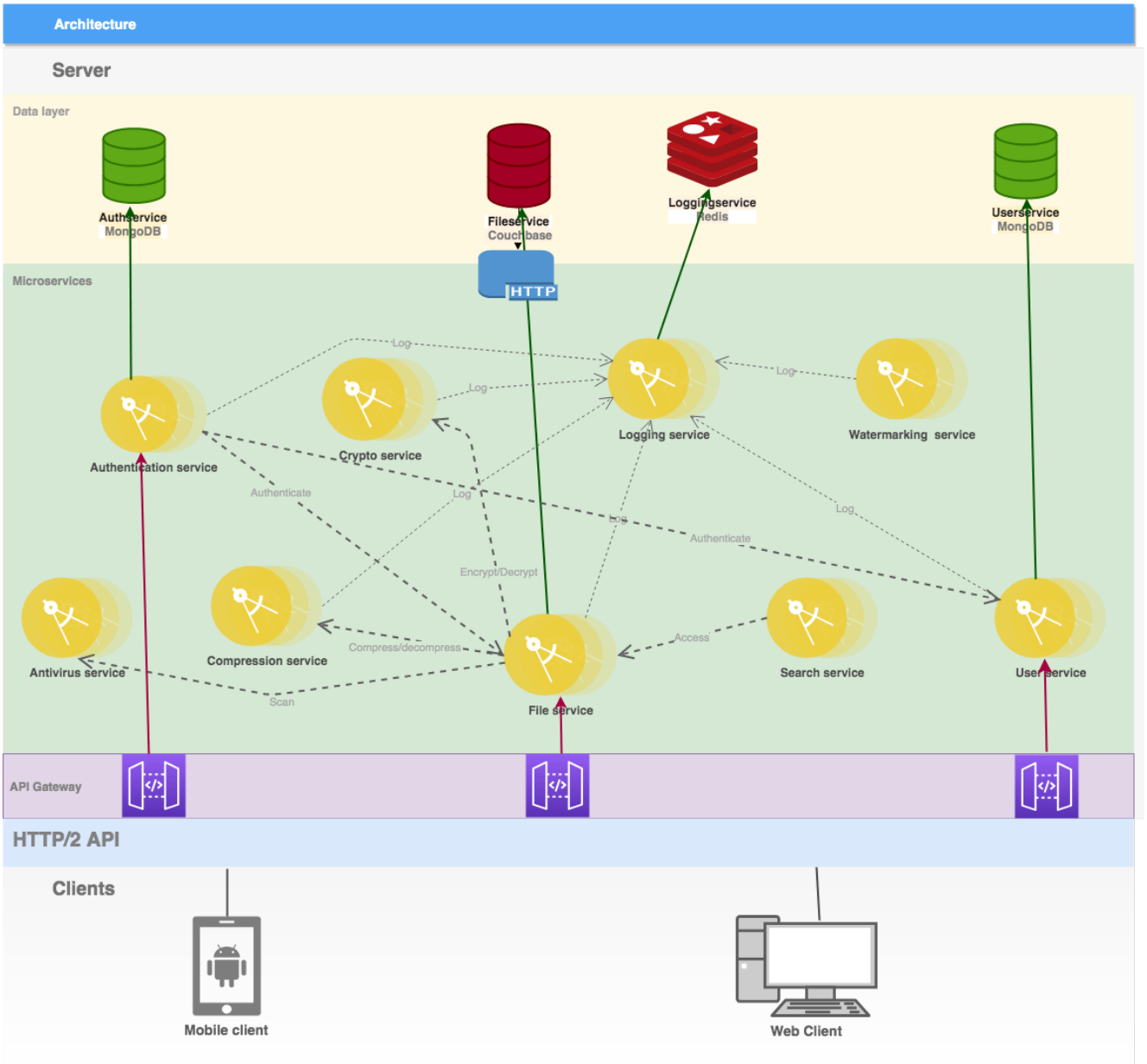

Figura 5.1: Arhitectura sistemului

De asemenea pentru dezvoltarea serverului au fost folosite 2 tipuri de baze de date: MongoDB şi Couchbase. Ambele fiind accesibile doar prin serverul dedicat. Pentru orchestrarea şi punerea în funcţiune a microserviciilor am folosit Docker si kubernetes. Pentru maparea API-urilor am folosit evnvoy şi Docker.

5.1.2 Orchestrarea microserviciilor

Microserviciile sunt o modalitate de a despărţi funcţionalităţile dintr-un sistem. Acestea ne oferă flexibilitatea de a scala funcţionalităţi specifice şi de a fi agili în livrarea produselor. După ce funcţionalităţile au fost separate în subsisteme dedicate, întrebarea următoare ar fi: cum să le ”lipim” la loc?

În procesul de stabilire a comunicării între servicii am avut o provocare destul de mare: să păstrez cuplarea la un nivel cât mai jos, în caz contrar ar putea apărea: multiple căderi, testare necalitattivă, dificultate crescută de înţelegere şi costuri crescute de consum.

5.1.3 Microserviciul de autentificare

Introducere

Arhitectură

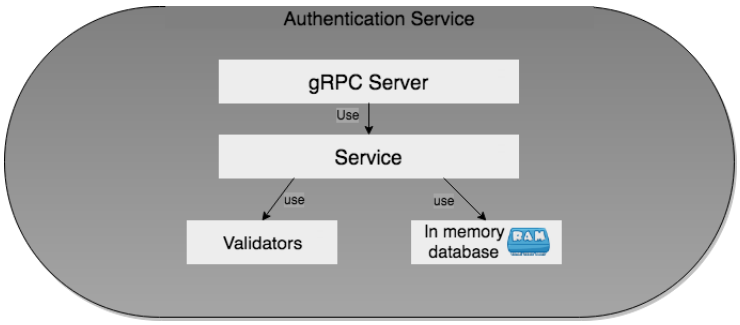

Figura 5.2: Arhitectura serviciului de autentificare

Rezultate obţinute

5.1.4 Microserviciul de criptare

Introducere

Arhitectură

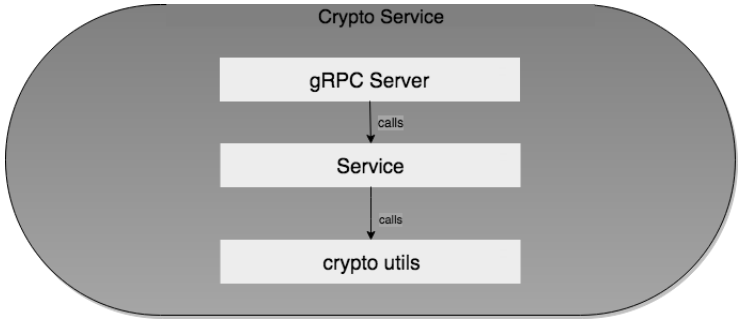

Figura 5.3: Arhitectura serviciului de criptare

Rezultate obţinute

5.1.5 Microserviciul de steganografie şi marcare

Introducere

Steganografia este ştiinţa ascunderii mesajelor, numele provenind de la cuvintele greseşti *steganos*, care înseamnă protejat, ascuns, şi *graphein*, care înseamnă a scrie.

Arhitectură

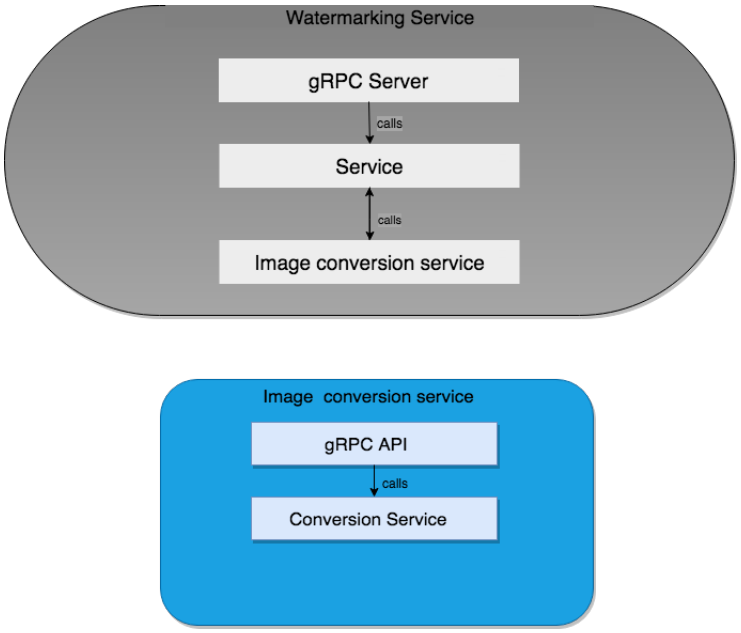

Figura 5.4: Arhitectura serviciului de marcaje

Aplicaţiile steganografiei

Rezultate algoritmi

5.1.6 Microserviciul de compresie

Introducere

Arhitectură

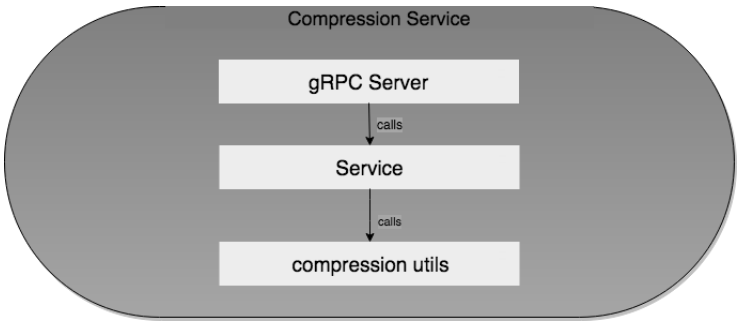

Figura 5.5: Arhitectura serviciului de compressie

Rezultate algoritmi

Microserviciul de compresie oferă 3 funcţionalităţi : compresie text, compresie PNG şi compresie JPEG.

Compresia de text este bazată pe zlib care foloseşte algoritmul DEFLATE, acesta permite utilizarea unui număr minim de resurse pentru a obţine un rezultat cât mai bun. Algoritmul este unul fără pierderi. Rezultatul bun este obţinut datorită folosirii arborilor de codificare Huffman, deorece se crează un arbore Huffman optimizat pentru fiecare block de date. Totuşi folosirea Huffman este recomandată mai mult pentru mesaje mici, deorece aceasta introduce pentru fiecare block nişte instrucţiuni de decompresie. Compresia este obţinută prin 2 paşi:

- Potrivirea şi înlocuirea datelor duplicate
- Înlocuirea simbolurilor cu altele noi bazate pe frecvenţa de apariţie

Dimensiune fişier text (octeţi)	Tip date	Dimensiune după compresie (Best)	Dimensiune după compresie (Default)	Dimensiune după compresie (Fast)	Dimensiune după compresie (Doar Huffman)	Dimensiune după compresie (No)
99	Repetat	48 48.48%	48 48.48%	68 68.69%	68 68.69%	115 116.16%
99	Unic	94 94.95%	94 94.95%	128 129.29%	94 94.95%	115 116.16%
2022	Repetat	183 9.05%	183 9.05%	187 9.25%	1112 55%	2038 100.79%
1769	Unic	710 40.14%	710 40.14%	756 42.74%	978 55.29%	1785 100.9%
10240	Repetat	230 2.25%	233 2.28%	295 2.88%	5368 52.42%	10256 100.16%
5277	Unic	1724 32.67%	1724 32.67%	1918 36.35%	2835 53.72%	5293 100.3%

Tabelul 5.1: Rata de compresie fişiere text

	Imagine mică	Imagine medie	Imagine mare	Imagine gigantă
Dimensiune Lăţime x Lungime	100 x 120	512 x 496	1024 x 1500	4000 x 4000
Dimensiune (octeţi)	36000	761856	4608000	48000000
Dimensiune compresie Best (octeţi)	236	1460	6646	56321

Raport dimensiune compresie Best	0.66 %	0.19 %	0.14 %	0.12 %
Timp execuţie compresie Best (s)	0,025696	0,460886617	2,69279	27,35
Dimensiune compresie Default (octeţi)	332	1825	7775	56321
Raport dimensiune compresie Default	0.92 %	0.24 %	0.17 %	0.12 %
Timp execuţie compresie Default (s)	0,025696	0,426049450	2,51625	26,207
Dimensiune compresie Fast (octeţi)	333	2057	9210	71688
Raport dimensiune compresie Fast	0.92 %	0.27 %	0.2 %	0.15 %
Timp execuţie compresie Fast (s)	0,015916	0,270366040	1,78344	16,9363
Dimensiune compresie No (octeţi)	36205	762756	4611603	48025301
Raport dimensiune compresie No	100.57 %	100.12 %	100.08 %	100.05 %
Timp execuţie compresie No (secunde)	0,004395	0,047150913	0,30246	2,97536

Tabelul 5.2: Rata de compresie PNG entropie mică

În cazul imaginilor cu o entropie mică, de exemplu cele care contin text sau au un procent de fundal foarte mare se obţine o rată de compresie foarte bună cu un factor cuprins între 20 şi 1000. Acest lucru contribuie la scăderea preţului de stocare per octet, însă introduce un cost mediu spre mare de timp pentru imaginile foarte mari.

	Imagine mică	Imagine medie	Imagine mare	Imagine foarte mare
Dimensiune Lăţime x Lungime	100 x 120	512 x 496	1024 x 1500	4000 x 4000
Dimensiune (octeţi)	36000	761856	4608000	48000000
Dimensiune compresie Best (octeţi)	36215	762931	4612658	48036298
Raport dimensiune compresie Best	100.6 %	100.14 %	100.1 %	100.08 %

Timp execuţie compresie Best(s)	0,055334	1,02189	5,769	65,3939
Dimensiune compresie Default (octeţi)	36215	762931	4612658	48036298
Raport dimensiune compresie Default	100.6 %	100.14 %	100.1 %	100.08 %
Timp execuţie compresie Default (s)	0,050935	0,989136	5,7694	62,03826
Dimensiune compresie Fast (octeţi)	36205	762756	4611603	48025301
Raport dimensiune compresie Fast	100.57 %	100.12 %	100.08 %	100.05 %
Timp execuţie compresie Fast (s)	0,030182	0,51095	3,0343	31,83958
Dimensiune compresie No (octeţi)	36205	762756	4611603	48025301
Raport dimensiune compresie No	100.57 %	100.14 %	100.08 %	100.05 %
Timp execuţie compresie No (secunde)	0,004997	0,048537	0,31703	3,02121

Tabelul 5.3: Rata de compresie PNG entropie mare

În cazul imaginilor cu o entropie mare compresia creşte dimensiunea fişierului prin adăugarea de metadate, iar dimensiunea datelor rămăne aceeaşi. În cazul unor imagini foarte mari de acest tip se poate introduce o întărziere de pănă la 35 de secunde fără a aduce impact asupra preţului de stocare. Însă aceste imagini se întâlnesc extrem de rar, aplicaţia ar fi afectată foarte puţin de acest fenomen.

5.1.7 Microserviciul de fisiere

Introducere

Arhitectură

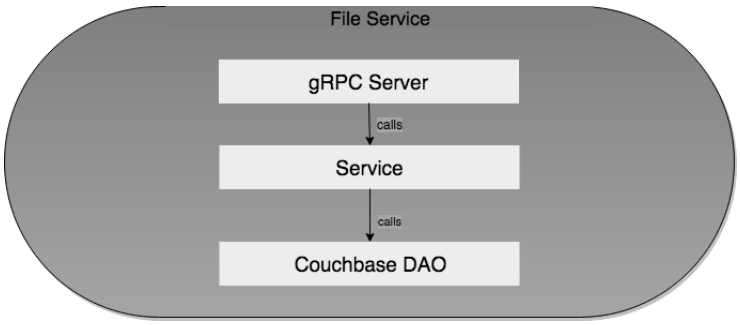

Figura 5.6: Arhitectura serviciului de fişiere

5.1.8 Microserviciul de utilizatori

Introducere

Arhitectură

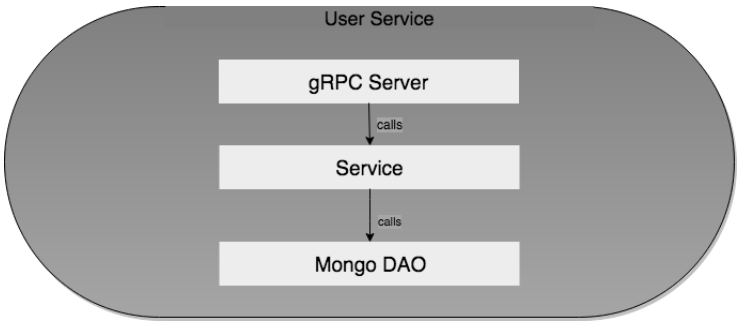

Figura 5.7: Arhitectura serviciului de utilizatori

5.2 Arhitectura aplicatiei web

5.2.1 Descriere generală

5.2.2 Descrierea componentelor

Capitolul 6

Testare şi Validare

Acest capitol conţine descrierea metodei de testare şi validate a sistemului. Sunt descrise toolurile care au fost folosite şi este analizatrocentul de acoperire a codului.

6.1 Testarea serverului

Pentru testarea serverului au fost scrise atât *Unit Teste*, cât şi *Teste de integrare şi interacţiune*. Pentru scrierea şi executarea testelor au fost folosite 2 *tool*-uri: *testing*, care este integrat în limbajul **Golang**, şi *testify*, care este o librărie *open-source* pentru testare în limbajul menţionat. Pentru o mai bună testare, înainte de dezvoltarea clientului *web*, s-au efectuat operaţii de testare utilizând fişierele *swagger* create pe baza API-ului serviciilor, şi utilizând tool-ul *Postman*.

În teste au fost acoperite toate cazurile de succes şi majoritatea cazurilor în care ar trebui să se producă o eroare, cum ar fi: parametri invalizi, parametri cu valori eronate sau precondiţii nesatisfăcute.

6.1.1 Reguli de testare în Golang

Pentru ca testele să poată fi executate automat, numele fiecărui fişier ar trebui să conţină sufixul *_test*. Fiecare metodă de test ar trebui să înceapă cu prefixul *Test*. Îndeplinirea acestor condiţii permite rularea automată a testelor într-un mediu de *Dezvoltare continuă*.

Scrierea unor teste bune nu este un lucru trivial, în multe situaţii sunt foarte multe cazuri de acoperit, ceea ce înseamnă cătrebuie scrise multe teste, **Golang** rezolvă această problemă prin adăugarea de *teste tabelare*, acestea permit crearea testelor imbricate, care pot fi rulate în paralel, asfel fiind mai uşor de detectat *cursele de date*. Acest tip de testare are un avantaj enorm deorece permite scrierea unui cod citibil şi foarte uşor de înţeles. Fiind integrate în limbaj, acest tip de teste pot fi create înainte de cod, imadiat după definirea semnăturilor de metode, ceea ce este ideal pentru o abordare *Test Driven Developement*.

De asemnenea, **Golang** permiite adăgarea de funcţii ajutătoare, a căror ordine şi frecvenţă de rulare poate fi condţionată, acestea pot fi precondiţii/postcondiţii globale sau locale.

6.1.2 Testarea serviciilor

Pentru fiecare serviciu au fost generate un numar de teste egal sau mai mare cu numărul de metode pe care le are serviciul, in fiecarea test au fost definite seturi tabelare de date pentru a acoperi cât mai multe cazuri. Testele au structura similară cu cel reprezentat în figura 6.1.

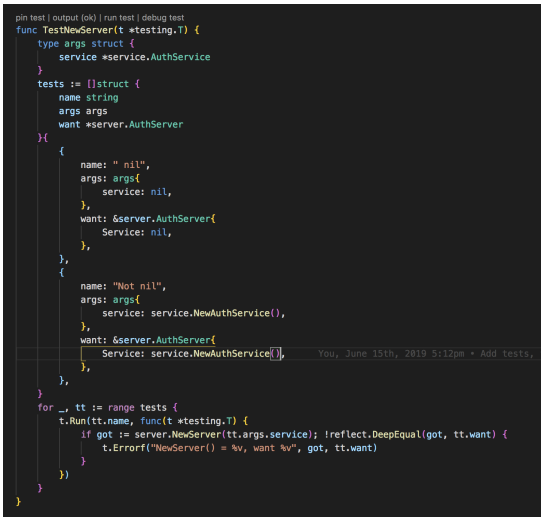

Figura 6.1: Exemplu Test

Rata de acoperire a codului a fost măsurată cu ajutorul unui *tool* care este integrat în limbaj, aceasta se numeşte **GoCover**. În tabelul 6.1 este reprezentată rata de aceprire cu teste a fiecărui serviciiu în parte, unele părţi nu au putut fi acoperite, acestea conţinând tratări de erori care vin din alte librării folosite, o situaţie de apariţie a acestora nu a putut fi simulată.

Tabelul 6.1: Rata de acoperire a testelor

Nume serviciu	Acoperire server(%)	Acoperire service (%)	Acoperire altele(%)

Testarea adecvată a avut un rol semnificativ în procesul de detecţie de erori sau comportament neaşteptat. Datorită testării au fost detectate probleme triviale de securitate, care puteau compromite identitatea şi datele utilizatorului.

6.2 Testarea clientului

postman si testtare manuala fe

Capitolul 7

Manual de Instalare şi Utilizare

7.1 Cerinţe preliminare

7.2 Instalare si configurare

Capitolul 8

Concluzii

8.1 Contribuţii şi rezultate obţinute

Aplicaţia obţinută oferă funcţionalităţi de încărcare şi descărcare a fişierelor, acesta fiind un sistem cu zero cunoţinţe despre datele stocate în sistem, un atacator nu poate să pună mâna pe datele utilizatorului în lipsa parolei de decriptare, aceasta nefiind stocată în sistem.

Prin adăugarea compresiei fişierelor text şi a imaginilor, implementate de mine, am obtinut reducerea coinsiderabilă a spaţiului de stocare în cazul fişierelor mari. De asemnea adăugarea compresiei a contribuit la creşterea securităţii aplicaţiei prin creştere nivelului de complexitate pentru decriptare.

8.2 Dezvoltări ulterioare

Operaţiile de criptare şi compresie sunt foarte costisitoare în cazul unor date mari ca volum şi cu rată de repetare mică, o soluţie care ar eficientiza procesul ar fi combinarea celor 2 operaţii în una singură. Acest efect se poate obţine prin adăugarea unor permutaţii pseudoaleatore în procesul de compresie al datelor[**?**].

Se pot eficientiza funcţionalităţile de încărcare şi descărcare a datelor prin adăugarea operaţiei de despărţire în blocuri, acestea fiind transmise alternativ, operaţiile de transfer prin reţea sunt cunoscute a fi cele mai încete operaţii de manipulare a datelor. O astfel de abordare ar putea reduce extrem de mult operaţiile de transmisie, dar şi cele de criptare şi compresie, datorită faptului că operaţiile pe blocuri ar putea fi efectuate în paralel, în loc de serial.

Bibliografie

[1] D. Reinsel, J. Gantz, and J. Rydning, "The digitization of the word from edge to core,' *IDC White Papper*, vol. 20, no. 18, pp. 1–27, 2018.

[2] P. Priyam, *Cloud Security Automation*, 2018.

[3] "An overview of monolithic vs microservices architecture (msa).' [Online]. Available: https://www.bmc.com/blogs/microservices-architecture/

[4] C. Richardson, "Pattern: Microservice architecture.' [Online]. Available: https://microservices.io/patterns/microservices.html

[5] J. Lewis and M. Fowler, "Microservices,' *ThoughtWorks*, vol. 1, no. 1, pp. 1–20, 2014.

[6] "Best cloud storage.' [Online]. Available: https://www.techradar.com/news/the-best-cloud-storage

[7] M. Borgmann, T. Hahn, M. Herfert, T. Kunz, M. Richter, U. Viebeg, and S. Vowe, *On the Security of Cloud Storage Services*, 2012.

[8] "Cloudme documentation.' [Online]. Available: https://www.cloudme.com/

[9] "Cloudwards - best cloud storage.' [Online]. Available: https://www.cloudwards.net/best-cloud-storage/

[10] "Crashplan review.' [Online]. Available: https://www.bestbackups.com/blog/5210/crashplan-review-2/

[11] "Icloud drive review.' [Online]. Available: https://www.cloudwards.net/review/icloud-drive/

Anexa A

Secţiuni relevante din cod

Lista figurilor

1.1 Creşterea volumului de date 2010-2025 4
1.2 Triada CIA 5

2.1 Arhitectura Client-Server 7
2.2 Exemplu de steganografie pe imagini 8

3.1 Arhitectura monolitică 13
3.2 Arhitectura bazată pe microservicii 15
3.3 Diferenţa dintre arhitecturi 16
3.4 CloudMe - interfaţa web 21
3.5 CloudMe - planuri de preţ 22
3.6 Dropbox - interfaţa web 23

4.1 Use cases 34
4.2 Golang 35

5.1 Arhitectura sistemului 38
5.2 Arhitectura serviciului de autentificare 39
5.3 Arhitectura serviciului de criptare 40
5.4 Arhitectura serviciului de marcaje 41
5.5 Arhitectura serviciului de compressie 41
5.6 Arhitectura serviciului de fişiere 45
5.7 Arhitectura serviciului de utilizatori 45

6.1 Exemplu Test 47

Lista tabelelor

3.1 Criterii evaluare sisteme similare 20
3.2 CloudMe Funcţionalităţi 21
3.3 CloudMe Platforme disponibile şi preţ 21
3.4 Dropbox Funcţionalităţi 23
3.5 Dropbox Platforme disponibile şi preţ 23
3.6 CrashPlan Funcţionalităţi 25
3.7 CrashPlan Platforme disponibile şi preţuri 25
3.8 ICloud Funcţionalităţi 26
3.9 ICloud Platforme disponibile şi preţuri 26
3.10 Google Drive Funcţionalităţi 27
3.11 Google Drive Platforme Disponibile şi Preţuri 27
3.12 OneDrive Funcţionalităţi 28
3.13 OneDrive Platforme Disponibile şi Preţuri 28
3.14 Funcţionalităţi 29
3.15 Sisteme de operare 29
3.16 Funcţionalităţi 30
3.17 Sisteme de operare 30
3.18 Comparaţie sisteme similare 31

5.1 Rata de compresie fişiere text 42
5.2 Rata de compresie PNG entropie mică 43
5.3 Rata de compresie PNG entropie mare 44

6.1 Rata de acoperire a testelor 47

Anexa B

Diagrame UML

Anexa C

Glosar

Termen	Definiţie
FTS	Full Text Search