

UNIVERSITATEA BABEȘ-BOLYAI
Facultatea de Științe Economice și Gestiunea Afacerilor
Informatică Economică

Lucrare de licență

Absolvent,
Lidia-Ancuța **TIMIȘ**

Coordonator științific,
Conf. univ. dr. Liana **STANCA**

2021

UNIVERSITATEA BABEȘ-BOLYAI

Facultatea de Științe Economice și Gestiunea Afacerilor

Informatică Economică

Lucrare de licență

Resurse creștine digitale pentru copii

Aplicație web

Absolvent,

Lidia-Ancuța **TIMIȘ**

Coordonator științific,

Conf. univ. dr. Liana **STANCA**

2021

Rezumat

Lucrarea intitulată Resurse creștine digitale pentru copii prezintă dezvoltarea unei aplicații web care are ca scop punerea la dispoziția copiilor interesați de cultura și principiile biblice, o gamă variată de resurse cu caracter creștin. Plecând de la premisa că în zilele noastre tehnologia influențează într-o mare măsură dezvoltarea personalității unui copil, am considerat utilă dezvoltarea acestei aplicații prin natura conținutului ei, fiind un instrument de promovare a virtuților creștine și o unealtă educativă pentru cei doritori. Utilizatorii vor avea posibilitatea de a interacționa cu sistemul prin funcționalitățile dezvoltate, urmărindu-se satisfacerea acestora prin interfața prietenoasă, resursele oferite și interactivitatea aplicației.

Cuprins

Introducere	1
1. Identificarea și descrierea problemei.....	2
1.1 Motivație.....	2
1.2 Context.....	4
2. Cerințe de sistem	9
2.1 Surse de cerințe	9
2.2 Elicitația cerințelor	9
2.2.1 Metoda Workshop-ului	9
2.2.2 Metoda Chestionarului.....	10
2.2.3 Metoda Cazurilor de Utilizare	12
2.2.4 Tabele de scenarii de utilizare	14
2.3 Documentarea cerințelor.....	17
2.3.1 Procese și activități	19
3. Modelul de dezvoltare.....	21
4. Proiectarea sistemului	23
4.1 Arhitectura sistemului.....	23
4.2 Baza informațională	27
4.3 Structura fizică a datelor	28
4.4 Procese și algoritmi	30
4.5 Tehnologii specifice.....	32
5. Implementarea sistemului	34
6. Testarea sistemului	39
Concluzii.....	44
Bibliografie	46
Anexe	47

Lista figurilor și tabelelor

Figuri:

Figura 1 – Diagrama Fishbone.....	5
Figura 2 – Diagrama de descompunere a obiectivelor	6
Figura 3 - Diagrama Pareto.....	13
Figura 4 – Diagrama cazurilor de utilizare	15
Figura 5 - Diagrama cazurilor de utilizare specifice administratorului	16
Figura 6 - Diagrama de stări	20
Figura 7 - Diagrama de activități	22
Figura 8 - Diagrama de Flux de Date	24
Figura 9 - Arhitectura Client – Server	25
Figura 10 - Diagrama de componente.....	26
Figura 11 - Diagrama bazei de date	29
Figura 12 - Bara de meniu	31
Figura 13 - Funcția de înregistrare.....	34
Figura 14 - Formularul de înregistrare.....	35
Figura 15 - Formular bară de căutare.....	36
Figura 16 - Pagina Home	36
Figura 17 - Pagina generică a unei poezii.....	37
Figura 18 - Secțiunea de cântece	38
Figura 19 - Caz de testare 1	39
Figura 20 - Eroare – Caz testare 2	41
Figura 21 - Rezolvare Eroare – Caz testare 2	42

Tabele:

Tabel 1 - Scenariu de utilizare 1.....	18
Tabel 2 - Caz de testare 1	40
Tabel 3 - Caz de testare 2	43

Introducere

Această lucrare de licență își propune să unească utilul cu plăcutul prin modul în care tehnologia se poate utiliza nu doar în sfera de activitate profesională și nu doar a indivizilor ajunși la vârsta maturității, ci poate modela un caracter cu valoare a unui copil. Era tehnologiei în care ne aflăm integrează lumea virtuală din ce în ce mai mult, nu doar în viața adulților, ci și a copiilor. Contextul pandemic a favorizat acest aspect, făcând din copii utilizatori activi ai mediului online. După cum afirma următorul proverb “Copilăria este o oglindă care reflectează în viața de mai târziu primele imagini” (Smiles, 1859), considerăm foarte importantă investiția făcută în copii pe toate planurile dezvoltării lor, fiind imperios necesar asigurarea unui cadru atât fizic, cât și online de educare în cercul unor principii fundamentale în viață.

Scopul acestei lucrări de licență este de a interpune printre tumultul informațiilor oferite prin intermediul internetului, un strop din ceea ce modelează caracterul unui copil, nu prin învățare ci printr-o modalitate de a desprinde el însuși principii din mediul creștin. În acest sens am realizat o aplicație web care oferă o gamă diversificată de resurse creștine destinate copiilor. În urma unei analize a site-urilor web care oferă această gamă de informații, am constatat că aria care acoperă această latură este una semnificativ restrânsă, aici fiind punctul de pornire în dezvoltarea acestei aplicații. Aceasta pune la dispoziția copiilor resurse text și audio care ajută la îmbogățirea culturii biblice și menținerea contactului cu mediul creștin, favorizând conservarea unui caracter bazat pe învățătura biblică. Limitarea posibilităților de deplasare și organizare a activităților creștine destinate copiilor în condițiile actuale au condus spre obținerea rezultatului reprezentat de aplicația realizată.

Lucrarea este structurată pe capitole după cum urmează: Capitolul 1 prezintă motivația și contextul în care s-a decis implementarea aplicației, Capitolul 2 abordează cerințele sistemului, Capitolul 3 vorbește despre modelul de dezvoltare ales, Capitolul 4 prezintă arhitectura sistemului, baza informațională, structura fizică a datelor, procese, algoritmi și tehnologii specifice, Capitolul 5 descrie implementarea sistemului, iar Capitolul 6 abordează tehnici de testare. Lucrarea se finalizează cu o concluzie și direcții viitoare de dezvoltare a sistemului.

1. Identificarea și descrierea problemei

Ideea dezvoltării unei aplicații web cu suport creștin destinat copiilor a luat ființă în momentul în care am conștientizat ceea ce afirma filosoful Aristotel “Plăcerea muncii aduce reușita acesteia” (Bostock, 1988). Considerând de o importanță vitală investiția culturii biblice în copii, am realizat o analiză a surselor de informare în acest sens oferite de piața virtuală dispuse într-o manieră cât mai sistematică și atractivă. Rezultatele acestui proces de analiză au reprezentat punctul de plecare în implementarea aplicației prin desprinderea unei concluzii care sugerează nevoia de dezvoltare a mediului creștin online destinate copiilor. Așadar, implementarea acestui tip de aplicație are rolul de a veni în sprijinul copiilor creștini care doresc să investească în creșterea pe plan spiritual prin timpul alocat mediului virtual. După cum am menționat și în introducerea lucrării, contextul pandemic actual a favorizat realizarea unei astfel de aplicații, întrucât a îngădit activitățile fizice în lumea din spatele ecranelor. Dacă pentru un adult petrecerea timpului online poate fi prejudiciabilă, cu atât mai mult reprezintă un teren periculos pentru cei lipsiți de prea multă experiență în utilizarea internetului (Kimberly, 1999).

1.1 Motivație

Un motiv important aflat la baza construirii acestei aplicații îl constituie crearea unui mediu care să ofere un plus de siguranță atât copiilor, care reprezintă utilizatorii sistemului, cât și părinților acestora. Desigur că motivul principal amintit la început este obținut prin unificarea mai multor motive secundare. Această abordare este ilustrată cel mai concis prin prisma Diagramei Fishbone. Aici se poate observa divizarea câmpului de motive în mai multe direcții. În cele ce urmează vom trata fiecare domeniu trecut în diagrama Fishbone, reprezentat prin forma dreptunghiului. Vom începe cu cauzele survenite din aria tehnologiei, cea mai familiară nouă. În cadrul analizei realizate pe această direcție identificăm deficiențe în ceea ce privește un sistem a resurselor bine structurat care să faciliteze îndeplinirea nevoilor utilizatorilor, prin diminuarea timpului acordat căutării de informație, punct următor în cauzele aferente domeniului tehnologiei. Una dintre cele mai prețioase resurse a vremurilor noastre fiind timpul (Jobs, 2005), construirea unei aplicații care reușeste să înglobeze atât resurse text cât și audio, aflate la un click distanță, este o reușită în ceea ce privește salvarea timpului destinat căutării informațiilor.

Rămânând tot pe linia tehnologiei, necesitatea unui mediu de învățare interactiv este o prioritate în dezvoltarea aplicației. Interactivitatea este urmărită atât în interfața aplicației, cât și în funcționalitățile oferite de aceasta, fiind un lucru foarte important de urmărit, mai ales că publicul țintă este reprezentat de copii, a căror atenție este mai ușor de captat prin interacțiune. De aici putem trece ușor la problema ridicată de acțiunea promovării unui site creștin, o altă cauză prezentă în diagrama Fishbone (Ishikawa, 1943). Generația care se ridică în zilele noastre are parte de creștere și dezvoltare în era tehnologiei, fiind cea mai la îndemână sursă de informare (Botnariuc, 2011) și promovare a informației, în cazul nostru a unui suport de învățare creștin. Considerăm că pentru susținătorii educației creștine, o astfel de aplicație este un instrument important de promovare și atragere a publicului creștin, și nu numai. Mai departe, putem aduce în discuție cauzele legate de lipsa interacțiunii copiilor cu mediul biblic.

Constrângerile create de situația pandemică în ceea ce privește desfășurarea în mod obișnuit a activităților destinate copiilor în cadrul bisericilor creștine constituie o altă cauză care a condus spre realizarea aplicației. În acest sens, resursele disponibile online sunt o substituție parțială a interacțiunii oferite în cadrul activităților respective, și o modalitate de continuitate a educației creștine. Mergând mai departe pe direcția educației, aducem în lumină alte cauze așezate în figura diagramei: lipsa materialelor educative suport disponibile online și selecția dificilă a unui conținut adecvat. O analiză atentă a tuturor cauzelor prezentate până în acest punct vom observa că toate acestea se rotesc în jurul unui cerc psihologic, deoarece avem ca utilizatori ai aplicației dezvoltate partea cea mai sensibilă a publicului – copiii, a căror personalitate este în dezvoltare (Robu, 2017). Din acest considerent, se necesită o atenție sporită asupra impactului adus de modul și timpul petrecut în mediul online. Siguranța oferită celor mici în interiorul sistemului dezvoltat oferă părinților un plus de încredere în influența internetului asupra intelectului copiilor lor.

Pasul următor în implementarea proiectului este stabilirea obiectivelor și ierarhizarea lor. În acest sens folosim Diagrama de descompunere a obiectivelor (Ciaca, 2021) care împarte obiectivele în trei mari direcții: accesul facil la o gamă variată de resurse, implementarea unui sistem interactiv și promovarea virtuților creștine. Fiecare dintre acestea se ramifică în obiective mai mici, care la rândul lor se împart pe mai multe ramuri.

Primul obiectiv care face referire la accesarea unei game variate de resurse se concretizează prin divizarea în două categorii: text și audio, dar nu înainte de a stabili obiectivele prin care se tratează modalitățile de utilizarea acestor resurse. Se urmărește deci ca utilizatorul să poată accesa

selectiv informația, fiind structurată pe categorii, să aibă opțiunea de a descărca anumite resurse și de a-i fi permis permanent accesul în aplicația web, nefiind limitat de un program de funcționare. Următorul obiectiv principal se ocupă de interactivitatea oferită de sistemul dezvoltat. Aceasta este propusă prin trei modalități: prin intermediul creării unui cont de utilizator care să-i ofere acesteia anumite facilități, printr-o secțiune de quizuri și prin crearea unei liste cu resursele preferate (Stepan, 2009). Aceste metode de interacțiune au ca scop menținerea unei relații de sprijin cu utilizatorul care să ajute la dezvoltarea și îmbunătățirea ulterioară a aplicației, urmărindu-se primordial satisfacerea nevoilor acestuia.

1.2 Context

Comportamentul sistemului este determinat de resursele care nu pot fi influențate în timpul dezvoltării acestuia, astfel pe această linie putem vorbi de un termen general care le definește și anume contextul sistemului (Ciaca, 2021). În cadrul acestei noțiuni dezbatem extragerea cerințelor, un pas foarte important aflat la baza dezvoltării unui sistem informatic. Un studiu realizat de Institutul Internațional de Analiză a Afacerilor afirmă că valorificarea cerințelor bine definite într-un mod eficient conduce la obținerea unui rezultat de succes, clienți satisfăcuți și dovedirea unui profesionalism crescut. Tratarea completă a cerințelor se face prin divizarea contextului în 4 fațete: fațeta subiect, fațeta utilizare, fațeta IT și fațeta dezvoltare (Ciaca, 2021).

Fațeta Subiect (Ciaca, 2021) se referă la conceptele relevante din contextul sistemului. Particularizat pe aplicația noastră vom scoate în relief baza informațională reprezentată de resursele oferite de sistemul creat, cât și utilizatorii aplicației. Resursele gestionate prin intermediul aplicației web se regăsesc în cadrul mai multor pagini organizate pe categorii de resurse. Vom avea de asemenea pagină destinată logării în sistem, funcționalitate care odată aplicată de utilizator deschide accesul nelimitat la toate resursele încărcate în aplicație, stocate mai întâi în baza de date aferentă, unde vom regăsi totodată și datele utilizatorilor care aleg să-și creeze un cont. Sfera din care fac parte materialele puse la dispoziție pe acest site web va fi cea creștină, mai precis cea destinată copiilor creștini care aleg să investească un timp educativ creștin în mediul virtual. Resursele vor fi în format text și audio, amintind aici și de existența unor teste de verificare a cunoștințelor din cultura biblică pe care copiii le vor putea accesa în regim logat.

RESURSE CREȘTINE DIGITALE PENTRU COPII

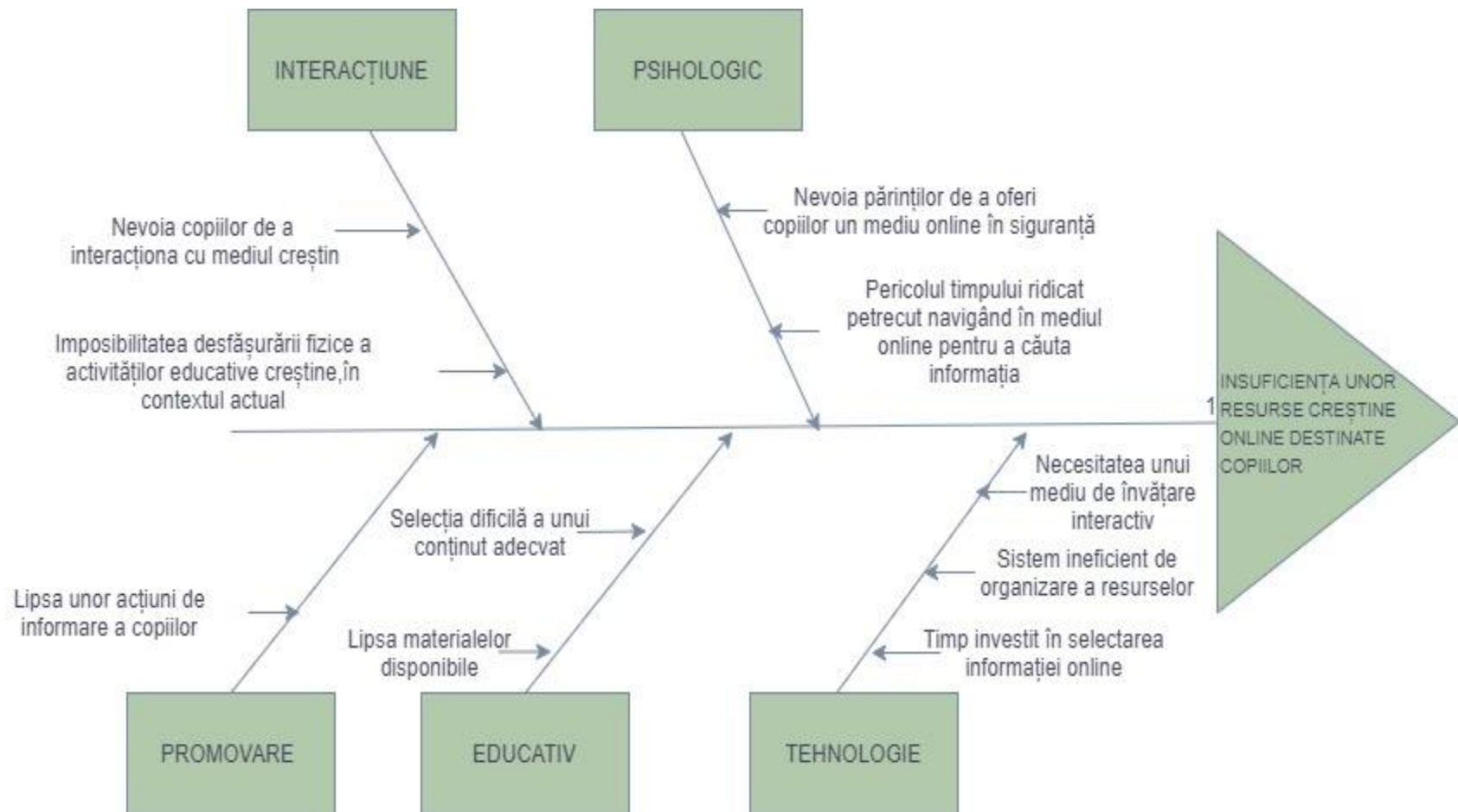


Figura 1 – Diagrama Fishbone, Adaptare după (Ishikawa, 1943)

RESURSE CREȘTINE DIGITALE PENTRU COPII

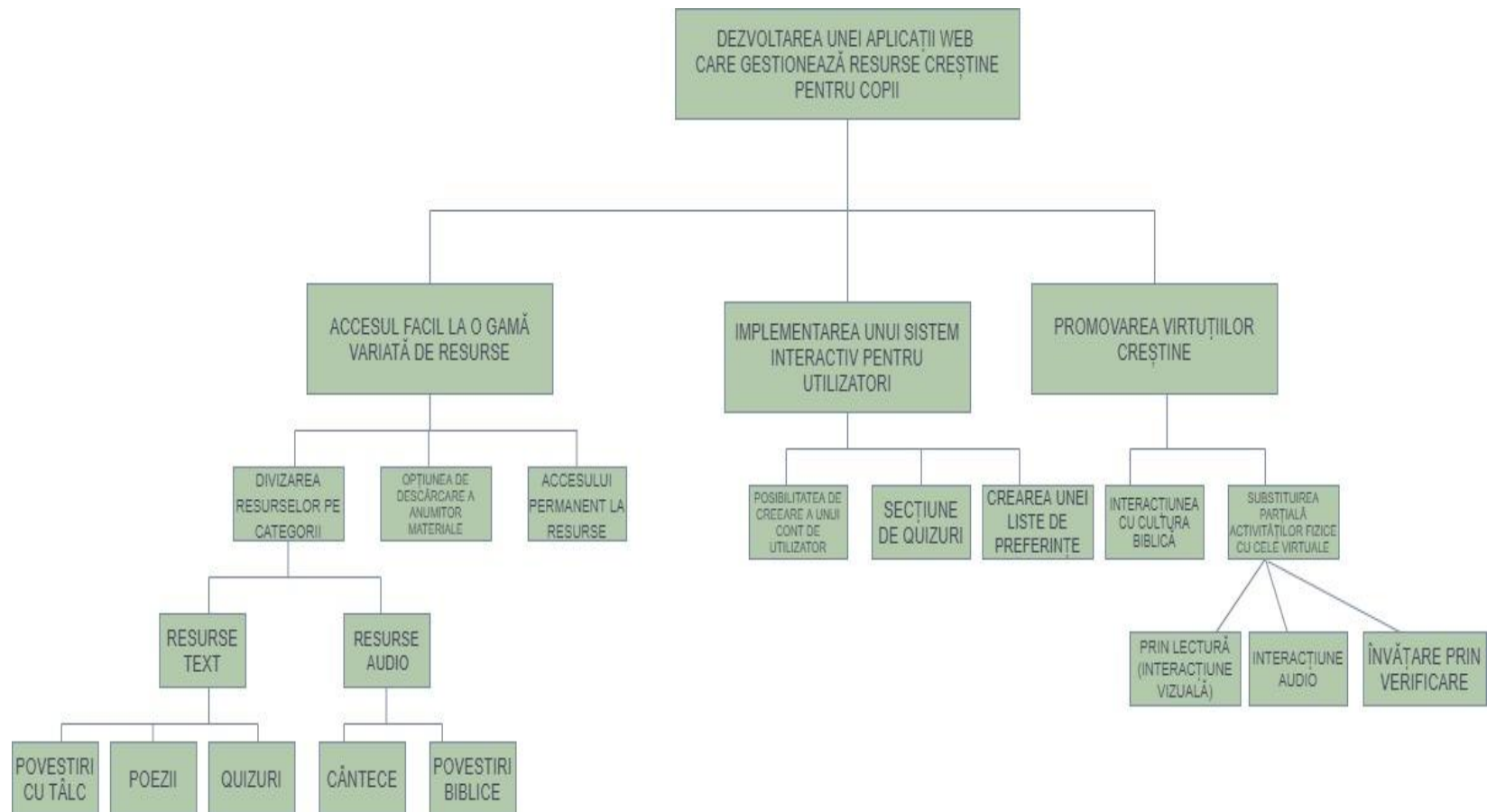


Figura 2 – Diagrama de descompunere a obiectivelor, Adaptare după (Ciaca, 2021)

Fațeta Utilizare (Ciaca, 2021) se preocupă de ceea ce sistemul oferă utilizatorilor prin funcționalitățile implementate și modul de utilizarea aplicației. Un aspect important amintit și mai sus este necesitatea logării pentru a beneficia de întreaga gamă de resurse. Pe partea audio regăsim povestiri biblice pe înțelesul celor mici, cântece interpretate de diferiți artiști contemporani, iar resursele text sunt reprezentate de povestiri cu tâlc în format pdf, o listă cu poezii și testele de verificare. Crearea unui cont de utilizator se realizează prin intermediul formularului din pagina de înregistrare, acțiune urmată de logarea în sistem cu datele aferente. În urma efectuării acestor pași, va fi deschis accesul spre orice resursă, cât și crearea unui liste cu resursele preferate. Tot în dependență de actul logării este și accesarea secțiunii de quizuri, care oferă oportunitatea ca utilizatorul să-și verifice cultura biblică prin intermediul unor întrebări, la finalul cărora va putea vedea rezultatul testului.

Fiind vorba despre o aplicație web, un pilon important de adus în discuție în contextul fațetei IT (Ciaca, 2021) este serviciul World Wide Web în cadrul căruia este utilizată aplicația. Fără de acesta, n-am putea vorbi de un site web, un public larg de utilizatori și de posibilitatea neîngrădită de accesare a site-ului creat. În ceea ce privește implementarea, vorbim de 2 direcții: frontend și backend. Interfața aplicației reprezintă partea de frontend, fiind partea din sistem care comunică cu utilizatorul. De această parte sunt responsabile limbajele de programare specifice și anume: HTML (HyperText Markup Language), CSS (Cascading Style Sheets) și Javascript. Este considerată partea mai ușor de implementat din procesul de dezvoltare. Trecând pe cealaltă direcție, vorbim despre procesele care au loc în spatele funcționalităților accesate din interfață. Concret, partea de backend constituie partea de programare din aplicație. Se încadrează aici comunicarea cu serverul, aplicația de interfață și legătura cu baza de date. Limbajul folosit pentru această parte este PHP, apărut în 1995 și dezvoltat de Rasmus Ledorf (Leodorf, 2002). Este un limbaj de scripting, cu cod open-source utilizat frecvent în dezvoltarea aplicațiilor web, de regulă integrat în codul HTML (Jason Lengstorf, 2009). Fiind un limbaj des folosit, este funcționalabil pe majoritatea sistemelor de operare, în cazul de față cel folosit fiind Microsoft Windows 10. Pentru partea de server vom folosi mediul de dezvoltare Wamp care conține automat serverul Apache și limbajul de programare MySQL pentru administrarea bazei de date. Aceasta din urmă stochează date despre resursele disponibile pe site, cât și despre utilizatorii acestuia, fiind populată atât prin

intermediul sistemului phpMyAdmin, cât și din interfața administratorului prin intermediul formularelor (Săulescu, 2003).

În contextul fațetei Dezvoltare (Ciaca,2021) vorbim despre dezvoltarea sistemului realizată prin metodologia Agile (Shore, Warden; 2007) Trăind într-o lume care se schimbă și evoluează într-un ritm alert, am considerat metodologia Agile cea mai potrivită, bazându-se pe schimbarea continuă, punct central în dezvoltarea software. Modelul de dezvoltare Scrum aplică principiile Agile și folosește sprint-urile – implementarea unui set de cerințe în iterații scurte, ca modalitate specifică. Conform abordării descrise în SCRUM Development Process (Schwaber, 1997), Scrum este o metodologie de management care poate fi aplicată atât unui sistem deja existent, cât și unui prototip urmat a fi dezvoltat. La fel ca și celelalte modele din cadrul Agile, prioritizează livrarea incrementală pentru obținerea rezultatelor la care clientul să aibă dreptul de schimbare. Această metodă face posibilă dezvoltarea pe pași a sistemului, lucru care reprezintă un factor motivator în munca depusă pe toată durata implementării. Este prioritar ca modelul de dezvoltare să creeze un context de lucru bine organizat în toate direcțiile, în cazul metodologiei Scrum cuvintele cheie care o descriu ar fi: intervale de timp fixate, roluri specifice, scopuri bine definite și rezultate concrete.

Odată înțeles contextul sistemului, vom continua în capitolul următor prin dezbateră cerințelor pe baza cărora va fi dezvoltat acesta.

2. Cerințe de sistem

2.1 Surse de cerințe

Înțelegerea cerințelor stakeholderilor constituie fundația unei soluții care să satisfacă nevoile beneficiarilor. Comunicarea dintre dezvoltatori și utilizatori trasează un drum sigur spre identificarea corectă a cerințelor, acțiune înfăptuită prin următoarele tehnici de extragere: metoda workshop-ului de cerințe, metoda chestionarului și metoda cazurilor de utilizare.

2.2 Elicitația cerințelor

Elicitația cerințelor este procesul de căutare, identificare și elaborare a cerințelor în cadrul dezvoltării unui sistem informatic. (Zowghi, Coulin; 2005). Este un proces complex care implică o varietate de tehnici, abordări și instrumente specifice. Implicați în acest proces sunt stakeholderii reprezentați de dezvoltatorii sistemului, utilizatori, experți în domeniu de activitate și managerul de proiect.

2.2.1 Metoda Workshop-ului

Am ales ca primă metodă de extragere a cerințelor metoda workshop-ului, considerând-o cea mai practică și eficientă din punct de vedere al obținerii unor cerințe clare. Această metodă este clar definită pentru a veni în sprijinul participanților de a dezvolta sugestii și direcții concise privind dezvoltarea viitorului sistem. (Westerlund, 2017). Ca mod de organizare presupune patru etape: preworkshop, sesiune, producție și continuare. Fiecare etapă are în vedere atingerea unor obiective clare care să conducă spre un rezultat dorit.

Prima etapă este cea de creare a contextului de desfășurare. După cum îi spune și numele presupune pregătirea atelierului de lucru prin organizarea și formarea echipei de participare, prezentarea subiectului de dezbateră și introducerea în tema sistemului care se dorește a fi dezvoltat. Stakeholderii participanți aparțin unor diverse grupuri sociale pentru a obține puncte de vedere cât mai diversificate. După parcurgerea etapei introductive vom intra efectiv în atingerea scopului acestei metode prin intermediul etapei de sesiune. Fiecare participant are posibilitatea de a-și expune propriile păreri, care ulterior vor fi puse cap la cap, analizate și recapitulate pentru a fi bine înțelese de către tot grupul de participanți. Un pas important în obținerea unui deznodământ favorabil este următoarea etapă, numită producție, care reprezintă mai exact rezumarea și sintetizarea ideilor prezentate de către fiecare individ. Aici se formează cerințe realizabile obținute

prin compactarea punctelor de vedere. Ultima etapă constă în prezentarea rezultatului final către client și trasarea direcțiilor de continuarea a procesului de dezvoltare.

În cazul nostru, fiind vorba despre o aplicație a cărei utilizatori reprezintă publicul larg al mediului online, cu precădere copiii interesați de materiale creștine, organizarea workshop-ului iese din granițele formalităților care în mod normal ar fi urmate într-o firmă. Cadrul este unul familiar, o atmosferă destinsă, prielnică deschiderii participanților și motivantă pentru exprimarea liberă a punctelor de vedere. Am ales un număr de zece participanți de diferite statuturi sociale: elevi, studenți, cadre didactice, programatori și casnici. Datorită diversificării ariei participanților am putut analiza mai multe idei diferite, atât cele provenite de la utilizatori activi ai mediului virtual, cât și de la personae cu fără prea multă experiență în utilizarea tehnologiei. Acest fapt a constituit un ajutor pentru obținerea unor direcții care privesc atât conținutul aplicației, cât și elemente ce țin de interfață, funcționalități, mod de utilizare.

Ideile s-au unificat în câteva puncte principale. Se consideră utilă dezvoltarea unei astfel de aplicații care să fie o sursă educativă, atât prin faptul că asigură copilului un mediu sigur de petrecere a timpului online, cât și un instrument de dezvoltare a caracterului creștin prin natura materialelor puse la dispoziție. Stakeholderii au fost de părere că este necesară diversificarea pe cât posibil a resurselor, pentru a spori interactivitatea sistemului, fiind de dorit să existe resurse text, audio, video, dar și funcționalități care să necesite un grad de concentrare mai ridicat, exemplificând prin existența unor teste de verificare a cunoștințelor biblice. Am concluzionat prin conturarea unor direcții de dezvoltare a unei aplicații cu caracter creștin care are ca scop valorificarea timpului acordat mediului virtual și promovarea educației creștine având la bază învățătura biblică.

2.2.2 Metoda Chestionarului

Inventat de polimatul Francis Galton în secolul al XIX- lea (Gillham, 2008), chestionarul este o metodă de cercetare și culegere a datelor prin intermediul unui set organizat de întrebări adresat unui public oricât de larg în scopul obținerii informațiilor bine delimitate. Aflat printre cele mai folosite instrumente de investigare, chestionarul oferă numeroase avantaje care motivează utilizarea lui, amintind dintre acestea: nu necesită resurse financiare, este familiar multor indivizi, permite analiza ușoară a rezultatelor, se poate adresa unui număr mare de respondenți (Bulzan,

Voinea; 2006). Având în vedere aceste considerente, am aplicat această tehnică ca metodă de obținere a cerințelor în cadrul dezvoltării aplicației noastre. Am formulat un număr restrâns de douăsprezece întrebări atât cu caracter general, cât și specific pentru a obține o imagine de ansamblu privind interesul și deschiderea publicului activ din mediul online asupra utilității unei aplicații web care oferă resurse creștine pentru copii. În urma completării chestionarului de către respondenți am obținut informații care conturează împreună necesitatea aplicației care se dorește a fi dezvoltată.

Am obținut răspunsuri de la diferite categorii de vârstă cu diferite statuturi sociale și niveluri de educație, dobândind astfel perspective variate. Ne-a atras atenția divizarea apropiată a statutelor sociale: 25% elevi, 30% studenți, 25% angajați și 20% casnici, care permite o viziune din puncte diferite de vedere. La fel putem spune și privind repartizarea pe nivelurile de studii. Dintre respondenți 45% au studii liceale, 32,5% studii universitare, 12,5% gimnaziale și 10% postliceale. Deși răspunsurile provin de la niveluri diferite de utilizare a tehnologiei, un procent de 47,5% afirmă că timpul petrecut online este de peste 3 ore. Din acest considerent, sistemul nostru reprezintă o metodă de valorificare a timpului ridicat petrecut online. Deși majoritatea respondenților, printre care și părinți, consideră că cea mai preferabilă formă de interacțiune a copiilor cu mediul creștin este cea fizică prin participarea la activitățile organizate în acest sens, totuși un procent de 37,5% consideră că situația pandemică ar trebui să asigure un alt mod de organizare utilizând tehnologia. Cât privește forma de utilizare a internetului pentru atingerea acestui scop, un procent surprinzător de 77,5% arată că este de dorit structurarea și disponibilitatea resurselor sub forma unei aplicații web, procentul celor care preferă să-și caute singuri informația este doar de 12,5%, restul consideră că acest aspect are o importanță scăzută sau chiar neconsiderabilă. Vorbind de aspectele care însoțesc un site web, peste 50% consideră importante și foarte importante diversitatea resurselor și interfața grafică prietenoasă. Având posibilitatea de a-și exprima liber preferințele asupra conținutului existent într-o aplicație de acest gen, majoritatea persoanelor chestionate au optat pentru resurse text, audio, video, dar și teste de verificare. Prin urmare, informațiile extrase prin metoda chestionarului au trasat necesitatea dezvoltării unei aplicații care să ofere copiilor posibilitatea de a accesa oricând și din orice loc resurse creștine diversificate, utilizând tehnologia în scop educativ și informativ.

Legat de această metodă de extragere a cerințelor aducem în discuție Diagrama Pareto, denumită după economistul italian Vilfredo Pareto (Ilieș, 2003). Această formă de reprezentare

grafică este un instrument decizional folosit pentru ierarhizarea obiectivelor în funcție de importanța acordată. Pe axa orizontală regăsim domeniile de interes ordonate descrescător în funcție de valoarea corespunzătoare de pe axa verticală, reprezentată de frecvența de apariție. Aceste categorii ierarhizate astfel au fost extrase din chestionarul analizat mai sus. Acest tip de grafic ne ajută în obținerea unei imagini de ansamblu asupra cauzelor care conduc spre realizarea sistemului. Pe primul loc în ierarhia cauzelor se află nevoia utilizatorilor de a dispune de o formă de organizare structurată a informației, urmată de rolul aplicației ca substituent al interacțiunii fizice în cadrul activităților creștine pentru copii. La polul opus, cu procentul cel mai mic de frecvență se află existența funcționalităților multiple. Păstrând această perspectivă asupra cerințelor de sistem ne vom îndrepta atenția în continuare asupra unei alte metode de extragere a cerințelor, metoda cazurilor de utilizare (Ciaca, 2021).

2.2.3 Metoda Cazurilor de Utilizare

Conform lucrării de specialitate, Guide to the Business Analysis Body of Knowledge (IIBA, 2005), metoda cazurilor de utilizare oferă o privire de ansamblu asupra funcționalităților și chiar a scopului pentru care sistemul a fost conceput. În cadrul acestei metode vom construi diagrama cazurilor de utilizare care va servi drept un instrument grafic alcătuit din actori, cazuri de utilizare și relațiile dintre aceștia. Principalul scop al acestei diagrame este să demonstreze aspectul dinamic al sistemului prin ceea ce ar trebui să facă într-un anumit caz de utilizare (Parker, 2009). Modul simplu de reprezentare grafică este o cale de comunicare pentru stakeholderi care ajută la înțelegerea practică aplicației (McLaughlin, 2006). Regăsim în afara chenarului cele 3 tipuri de actori ai sistemului, sub forma unor omuleți: administrator, utilizator logat și vizitator, iar în interior scrise în interiorul ovalelor sunt cazurile de utilizare aferente fiecărui actor. Relațiile care apar în cazul de față sunt de două tipuri: relații de asociere și relații de incluziune (Cockburn, 2001). Relații de asociere există între actori și fiecare caz de utilizare, iar relații de incluziune avem între cazurile de utilizare, spre exemplu cazul de utilizare al creării unei liste de preferințe include autentificarea în sistem. Actorul care îl reprezintă pe administratorul sistemului are rolul de a gestiona resursele prin efectuarea operațiilor CRUD, pentru care este necesară mai întâi logarea în aplicație cu un cont specific acestui rol. Utilizatorul care a efectuat pasul autentificării beneficiază de toate funcționalitățile existente și accesul nerestricționat spre orice resursă. Putem vorbi aici de crearea unei liste de preferințe, descărcarea unei resurse sau rezolvarea unui quiz. Toate aceste

RESURSE CREȘTINE DIGITALE PENTRU COPII

funcționalități nu apar și în dreptul unui vizitator care nu deține un cont creat, acesta având doar posibilitatea de răsfoire a resurselor și de a obține statutul unui utilizator cu drepturi depline prin completarea formularului de înregistrare.

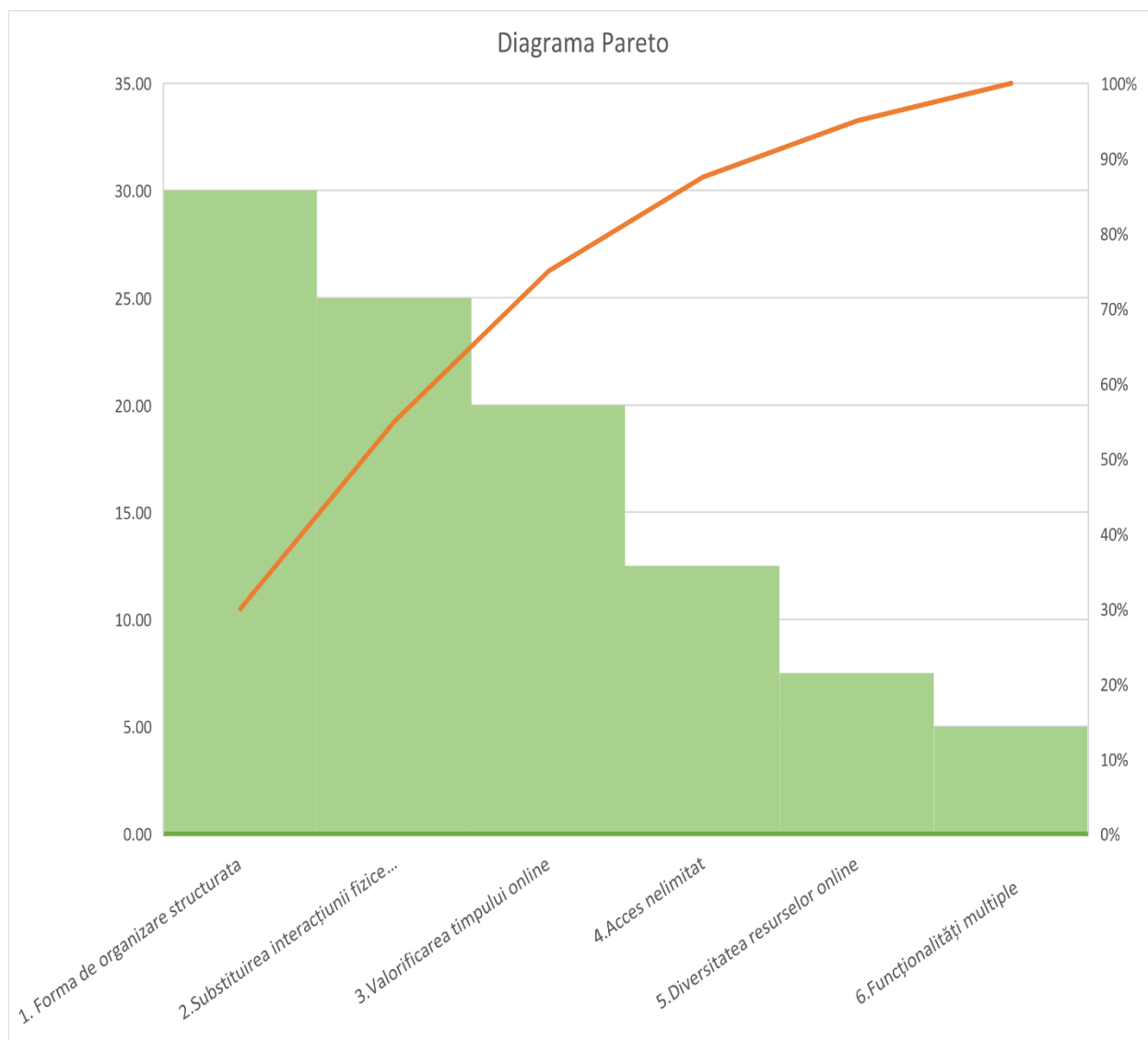


Figura 3 - Diagrama Pareto, Adaptare după (Pareto, 1917)

O înțelegere mai aprofundată o vom avea analizând un caz de utilizare particular pentru care vom construi o altă diagramă de acest tip, Figura 5. Se poate observa din această figură

atribuțiile avute de administrator în cazul sistemului. Incluse în administrarea aplicației sunt cazurile de utilizare reprezentate de operațiile de inserare, ștergere și modificarea resurselor. Diagrama se dezvoltă prin detalierea operației de inserare, care presupune alegerea unui titlu, a unui autor dacă este cazul și specificarea sursei sau introducerea unui conținut.

2.2.4 Tabele de scenarii de utilizare

Scenariile de utilizare reprezintă exemple practice pentru diferite variante de utilizare a sistemului prin precizarea pașilor care trebuie urmați în cadrul unui caz de utilizare specific (Coleman, 1998). Scenariu de utilizare 1 și Scenariu de utilizare 2 (vezi Anexa 2), descriu două modalități diferite de funcționare a aplicației concepute pentru 2 actori diferiți. Primul scenariu de utilizare face referire la dreptul utilizatorului de a descărca anumite resurse din aplicație. Este un scenariu de interacțiune datorită pașilor care trebuie făcuți în efectuarea lui. În scurte cuvinte, obiectivul urmărit este deținerea unei resurse în afara aplicației pentru a putea fi accesată în lipsa conexiunii la internet a utilizatorului. Această acțiune se poate desfășura doar în urma îndeplinirii unei precondiții care presupune logarea în sistem, trecând la urmarea pașilor aferenți scenariului. Primul pas este accesarea aplicației printr-un browser web unde sistemul îi oferă posibilitatea de autentificare. Completarea formularului de logare constituie următorul pas succedat de disponibilitatea de access a tuturor resurselor. Se continuă cu căutarea resursei dorite și descărcarea acesteia prin apăsarea butonului de descărcare. Scenariul se încheie cu existența resursei pe dispozitivului utilizatorului. Postcondiția ne spune că utilizatorul poate descărca ulterior și alte resurse. Scenariul prezentat este în legătură și cu un alt scenariu reprezentat de acțiunea de autentificare. Al doilea scenariu de utilizare este operația de ștergere a unei resurse prin interfața administratorului. Prin autentificarea cu contul de administrator, acesta are posibilitatea de a alege o resursă și de a o elimina din baza de date a aplicației, prin urmare și din interfață, fapt înfățișat prin postcondiție. Pașii urmați sunt în ordinea următoare: logarea cu contul de admin, căutarea resursei, apăsarea butonului de ștergere și obținerea rezultatului prin care se observă dispariția resursei din listă. Actorii implicați în acest caz sunt administratorul și sistemul, de unde rezultă caracterul de interacțiune al scenariului.

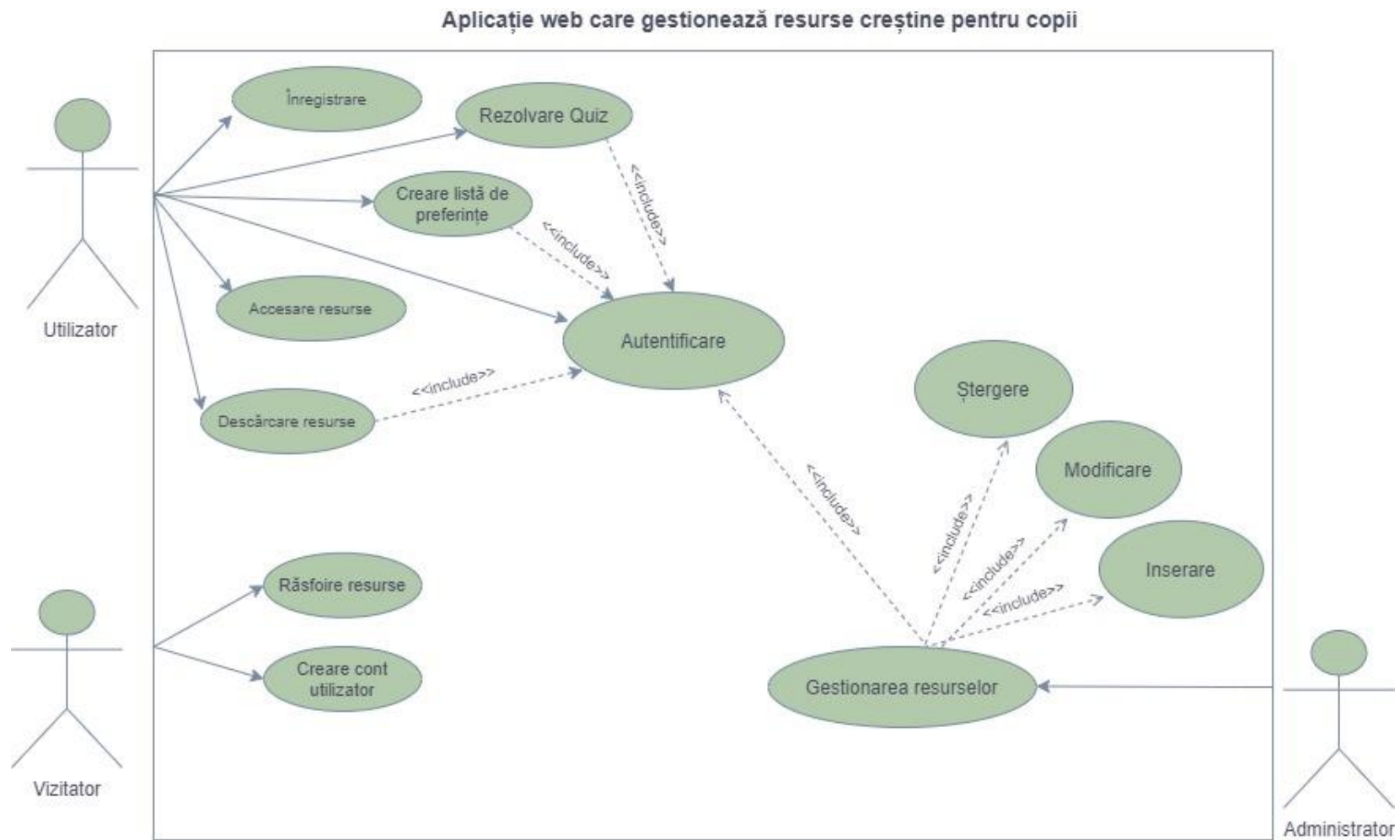


Figura 4 – Diagrama cazurilor de utilizare, Adaptare după (Ciaca, 2021)

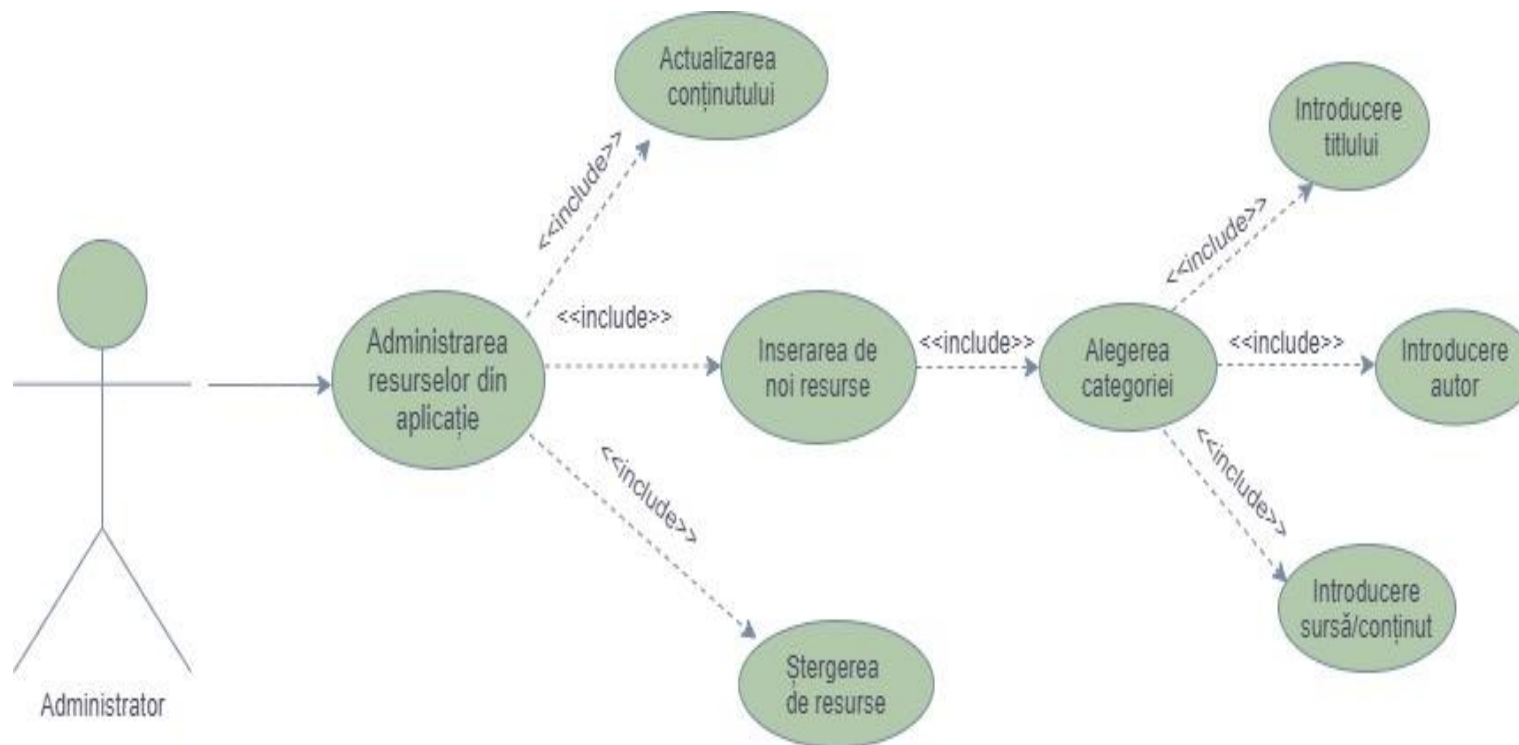


Figura 5 - Diagrama cazurilor de utilizare specifice administratorului, Adaptare după (Ciaca, 2021)

2.3 Documentarea cerințelor

Documentarea cerințelor se împarte în două direcții. Prima este partea funcțională în care sunt cuprinse interacțiunile actorilor cu sistemul, obiective urmărite, fluxuri, procese, activități și evenimente realizate. A doua direcție se preocupă de modul în care se produc și se consumă datele în sistem (Ciaca, 2021). Analizând cele două perspective, vom obține o clasificare mai detaliată a cerințelor, prin urmare vom continua studiul pe categorii de cerințe.

O ramificare a cerințelor poate fi făcută în cerințe funcționale și cerințe non-funcționale (Ciaca, 2021). Cerințele funcționale surprind comportamentul preconizat al sistemului. Acest comportament poate fi exprimat prin servicii, sarcini sau funcții pe care sistemul trebuie să le îndeplinească. (Malan, 1999). În acest context intră în discuție modul în care sistemul răspunde stimulilor externi, cum ar fi procese de calcul, procesarea și manipularea datelor. Recunoaștem aici rolul cazurilor de utilizare discutate la punctul anterior. Prin intermediul acestora se generează practic această categorie de cerințe. Reprezentarea grafică se corelează cu Diagrama Cazurilor de Utilizare, unde se găsesc concret punctele pe care sistemul trebuie să le atingă. Pentru a beneficia în totalitate de funcționalitățile sistemului, este necesară îndeplinirea cerinței de înregistrare apoi autentificare în sistem. Fără acestea nu se poate trece mai departe la descărcarea resurselor care au această disponibilitate. Contactul cu administratorul, lista de preferințe sau rezolvarea quizurilor sunt cuprinse, de asemenea, în cercul funcționalităților sistemului.

Cerințele non-funcționale reprezintă constrângeri la nivel de implementare asupra cerințelor funcționale (Pescaru, 2004). Acestea fac referire la calitatea sistemului în ceea ce privește modul de dezvoltare și utilizare. Referitor la aplicația noastră trebuie să se încadreze în limitele care privesc timpul, bugetul, performanța, siguranța și lista continuă cu o serie de atribute. Faptul că aplicația suport este de tip web apare o constrângere de disponibilitate, astfel că aceasta să poată fi accesată la orice oră, de pe orice dispozitiv care conține un browser. Nu trebuie să existe restricții de acces depinzând de numărul utilizatorilor în același timp sau de browserul folosit. Securitatea aplicației privind integritatea resurselor este asigurată prin protecția contului de administrator, singurul care poate executa operațiunile CRUD asupra resurselor.

RESURSE CREȘTINE DIGITALE PENTRU COPII

Secțiune	Conținut
Identificator	A.R.C
Nume	Aplicație web care gestionează resurse creștine pentru copii
Autor	Timiș Lidia-Ancuța
Versiunea	V.0.1
Prioritate	Medie
Severitate	Medie
Responsabil	Timiș Lidia-Ancuța
Scurtă descriere	După logare, utilizatorul are acces la toate resursele sistemului, având posibilitatea de a descărca unele materiale pe dispozitivul
Tip de scenariu	Scenariu de interacțiune
Obiectiv	Accesul în mod offline la anumite resurse cu opțiunea de descărcare.
Actori	Utilizator și sistem
Precondiție	Existența unui cont de utilizator și logarea în sistem cu contul aferent.
Postcondiție	Utilizatorul are permisiunea de a descărca ulterior și alte resurse.
Pașii scenariului	<ol style="list-style-type: none"> 1. Utilizatorul accesează aplicația web printr-un browser. 2. Sistemul pune la dispoziția acestuia resursele disponibile și opțiunea de logare. 3. Utilizatorul se loghează în sistem pe baza contului existent. 4. Aplicația ridică restricțiile de access pe baza logării și oferă în întregime funcționalitățile existente. 5. Utilizatorul caută materialul dorit pentru descărcare. 6. Găsirea cu succes este urmată de acțiunea propriu-zisă de descărcare a resursei dorite. 7. Utilizatorul are disponibilă resursa offline și poate continua navigarea în aplicație
Calitate	Procesul de descărcare trebuie executat fără întârzieri de timp și fără alte restricții, mai puțin logarea în sistem.
Relații cu alte scenarii	Autentificarea utilizatorului în aplicație.

Tabel 1 - Scenariu de utilizare 1, Adaptare după (Ciaca, 2021)

2.3.1 *Procese și activități*

Tranzițiile prin care trece un obiect când asupra lui este exercitat un anumit stimul alcătuiesc împreună diagrama de stări (Ciaca, 2021). Această formă de reprezentare grafică pune în lumină înlănțuirea stărilor suferite de un obiect de la un moment inițial la unul final. Este una din cele mai populare metode prin care se descrie comportamentul dinamic al sistemelor complexe (Larman, 2004). Înțelegerea acestui aspect teoretic o vom realiza printr-un model suport de rezolvarea unui quiz din cadrul aplicației noastre. Demararea întregului proces are la bază logarea în aplicație, urmând apoi enumerarea evenimentelor, încadrate în dreptunghiuri și relaționate între ele prin tranzițiile reprezentate de săgeți. În altă ordine de idei, evenimentele sunt rezultatele efectuării acțiunilor asupra quizului care se dorește a fi rezolvat. Prin selectarea unui anumit răspuns din cadrul unei întrebări se ajunge la consemnarea vizuală a răspunsului ales, reprezentată prin marcarea acestuia. După derularea tuturor evenimentelor se ajunge la starea finală a quizului, de afișare a punctajului obținut.

Continuăm studiul pe documentarea grafică din perspectiva funcțională prin prezentarea diagramei de activități, instrument aflat în legătură cu cazurile de utilizare. Diagrama de activități modelează dinamica unui proces sau obiect reflectând modul de interacțiune a sistemului cu mai mulți actori (Rumbaugh, 2001). Particularizată pe sistemul nostru, diagrama de activități prezintă în paralel activitățile realizate de utilizator și sistem în procesul de descărcarea unei resurse. La fel ca în cazul diagramei anterioare, stările care pot fi activități sau acțiuni sunt încadrate în dreptunghiuri iar tranzițiile, reprezentând relația dintre 2 stări, sunt ilustrate grafic prin săgeți. Procesul are o stare inițială și o stare finală redate prin cele 2 tipuri de cercuri (Fowler, 2004). În cazul nostru, diagrama de activități prezintă în plus ramificațiile, folosite pentru modelarea deciziilor în anumite momente. Un exemplu în acest sens poate fi observat în apropierea stării finale când după descărcarea resursei dorite se poate reveni în punctul de căutare unei noi resurse sau se poate continua prin ieșirea din aplicație. Analog, se poate reprezenta și un alt caz de utilizare prin acest tip de diagramă care facilitează înțelegerea modului în care sistemul răspunde stimulilor exteriori care se exercită asupra lui.

Odată lămurite aceste aspecte, în capitolul următor vom aborda modelul de dezvoltare ales pentru realizarea aplicației care gestionează un set de resurse creștine pentru copii.

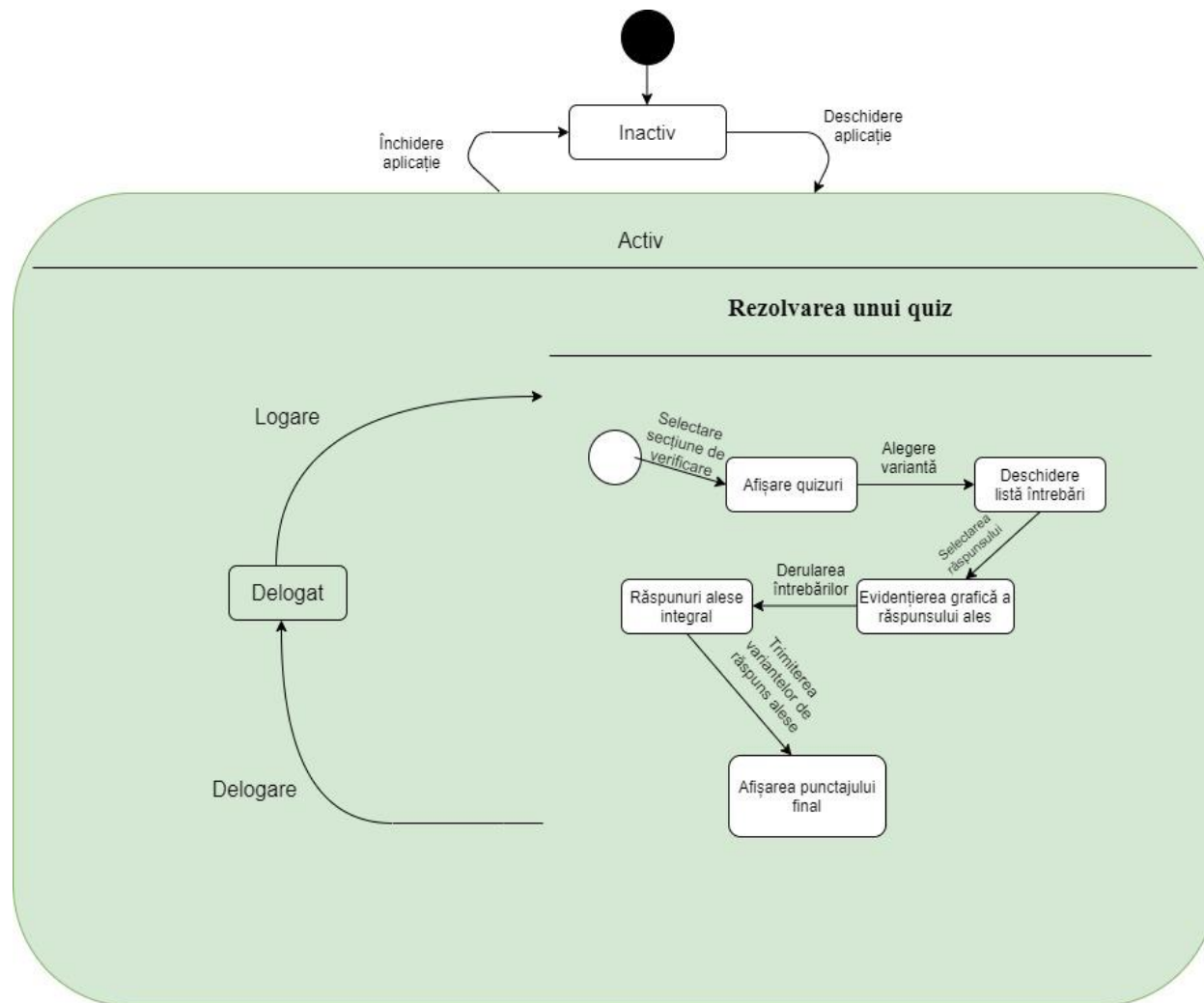


Figura 6 - Diagrama de stări, Adaptare după (Ciaca, 2021)

3. Modelul de dezvoltare

În realizarea acestui proiect, am ales ca și mediu de dezvoltare framework-ul Scrum (Sutherland, 2010) făcând parte din rândul metodologiilor agile. Această binecunoscută și folosită metodă se remarcă prin abordarea organizată, structurată și eficientă a procesului de implementare, dar și prin transparența de care se dă dovadă pe tot parcursul proiectului (Rubin, 2010).

Dezvoltat de Ken Schwaber și Jeff Sutherland, framework-ul Scrum se face cunoscut în 1995 atunci când apare public documentul The Scrum Development Process, unde se prezintă pe larg în ce constă această metodă (Schwaber, Sutherland, 1995). Specifice acesteia sunt sprint-urile, etapele în care este împărțită dezvoltarea proiectului, reprezentând intervale scurte de timp în care se rezolvă cerințe punctuale. În fiecare zi, pe durata unui sprint în cadrul echipei de dezvoltare se organizează întâlniri scurte – Daily Scrum, de 10-15 minute care au rolul de a organiza și planifica într-un mod eficient sarcinile din ziua respective. Se observă un mod de organizare eficient și bine determinat pentru fiecare membru al echipei (Sutherland, 2010).

Un rol deosebit de important în metoda Scrum îl are utilizatorul implicat continuu în dezvoltarea proiectului (Schwaber, Sutherland; 1995). În cadrul aplicației noastre, implicarea utilizatorului a fost vizibilă încă dinaintea începerii procesului de implementare, prin trasarea cerințelor sistemului, fiind prezent activ pe întreaga durată a dezvoltării. În ceea ce privește organizarea echipei de proiectare, este împărțită pe trei niveluri. Primul și cel mai important este Product Owner-ul, cel care contruiește puntea principală între stakeholderi și produsul care se dorește a fi realizat. Responsabilitatea sa este de a coordona munca echipei, de a superviza la fiecare pas desfășurarea activităților și de a valorifica rezultatele fiecărui sprint. Un rol important de asemenea îl are Scrum Master-ul a cărui arie de activitate acoperă metodologia de implementare (Schwaber, Sutherland, 1995). Acesta se asigură că echipa respectă planul de lucru și dezvoltă corect sistemul, respectând normele trasate. Restul membrilor din echipa Scrum sunt reprezentați de echipa de dezvoltare, mână de lucru a proiectului. Echipa este formată din 5-9 membri cu competențe tehnice și abilitatea de lucru în echipă. Această metodă de dezvoltare se pliază pe cerințele aplicației noastre care gestionează o gamă largă de resurse creștine, printr-o livrare incrementală și o implicare continuă utilizatorului.

În capitolul următor vom discuta despre proiectarea sistemului prin abordarea prelucrării datelor la nivel de sistem.

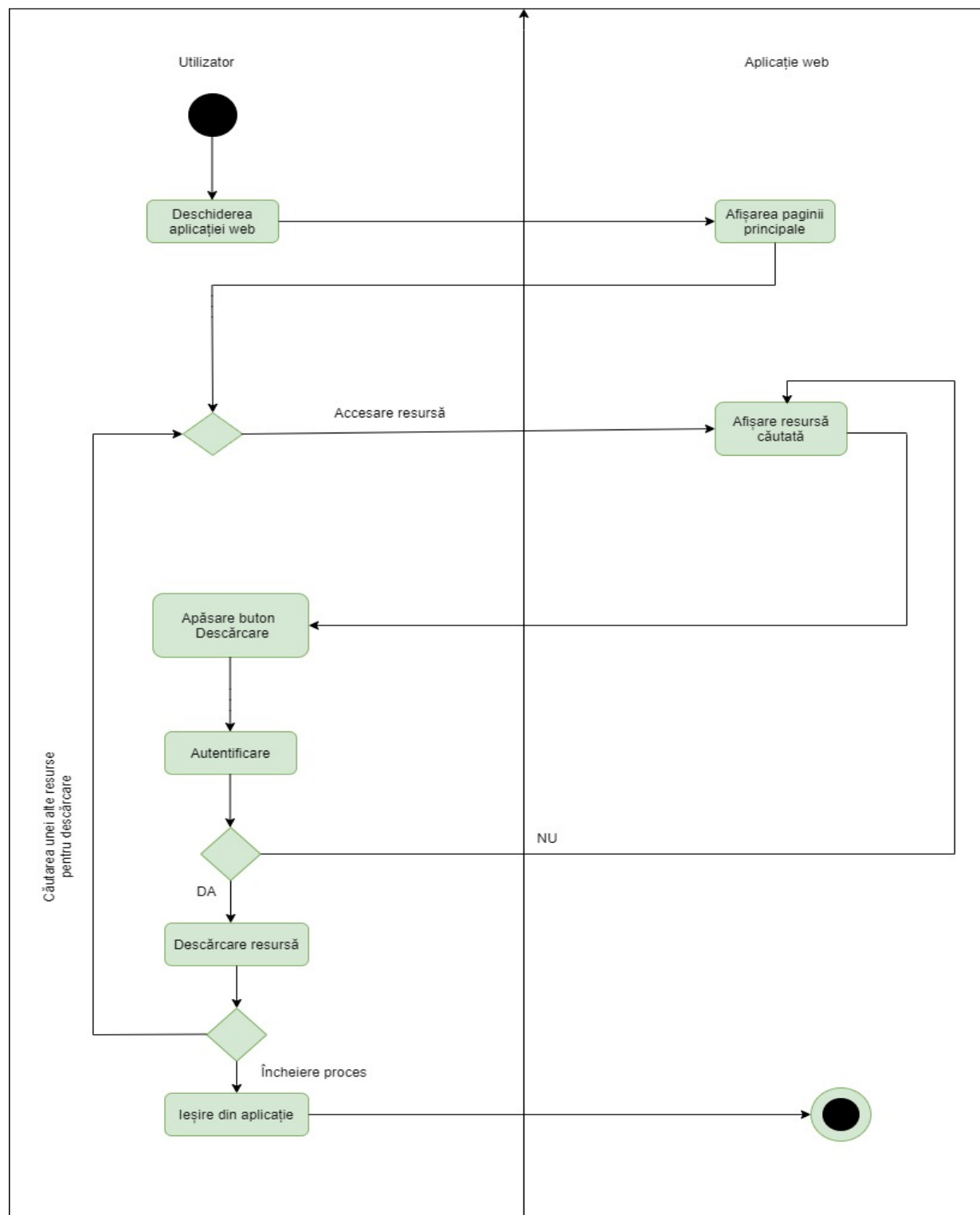


Figura 7 - Diagrama de activități, Adaptare după (Ciaca, 2021)

4. Proiectarea sistemului

Din perspectiva stocării datelor și a complexității sistemului avem o clasificare a sistemelor în două categorii: sisteme informatice centralizate și sisteme informatice distribuite (Moldovan, 2006). Pentru sistemul nostru, datorită dimensiunii și complexității reduse am ales abordarea centralizată din următoarele considerente care ne-au avantajat: control efectiv asupra procesului de dezvoltare, securitatea datelor și promovare facilă la nivelul întregului sistem (Stepan, 2009).

În acest context ne vom folosi de Diagrama de Flux de Date pentru a dobândi o înțelegere clară a modului în care datele de intrare sunt prelucrate în interiorul aplicației pentru obținerea datelor de ieșire. Această diagramă ne oferă o viziune asupra pașilor și a datelor implicate într-un proces din sistem (Yourdon, 1988). Totodată este un mod grafic de a arăta scopul aplicației, ceea ce oferă sistemul, dar nu și cum oferă (Ambler, 2007). Elementele componente au fiecare un rol bine definit și le vom prezenta pe scurt în cele ce urmează. Procesul este o activitate prin care se transformă datele de intrare în rezultat, fiind ilustrat grafic printr-un cerculeț. Sursa se regăsește în interiorul unui dreptunghi și simbolizează un obiect din mediul extern care relaționează cu sistemul. Între 2 linii orizontale se notează depozitul de date, după cum îi spune și numele indică stocarea datelor. O ultimă componentă este fluxul de date dintre 2 procese, un proces și un depozit de date sau o sursă și un proces, simbolizându-se printr-o săgeată curbată (Ciaca, 2021). Particularizată pe sistemul nostru, diagrama de flux de date arată trei ramuri de funcționare a aplicației pe direcția prelucrării datelor. Se poate observa în primul rând sursa reprezentată de utilizator care comunică cu sistemul. De aici am dezvoltat diagrama prin ilustrarea a trei procese din care s-au obținut rezultate. Procesul autentificării presupune fluxul credențialelor unui utilizator din care rezultă statutul de utilizator logat în aplicației. Acest proces se leagă de celelalte două reprezentate în diagramă prin condiția autentificări pentru accesarea funcționalităților.

4.1 Arhitectura sistemului

Arhitectura unui sistem informatic se definește printr-un set de modele conectate între ele care descriu elementele esențiale ale sistemului (Sidorova, 2004). Arhitectura rezultă din totalitatea componentelor și interconexiunilor dintre ele, astfel că acest concept se folosește nu doar în dezvoltarea sistemului, ci și mentenanța acestuia (Voorhoeve, 2004). În implementarea aplicației

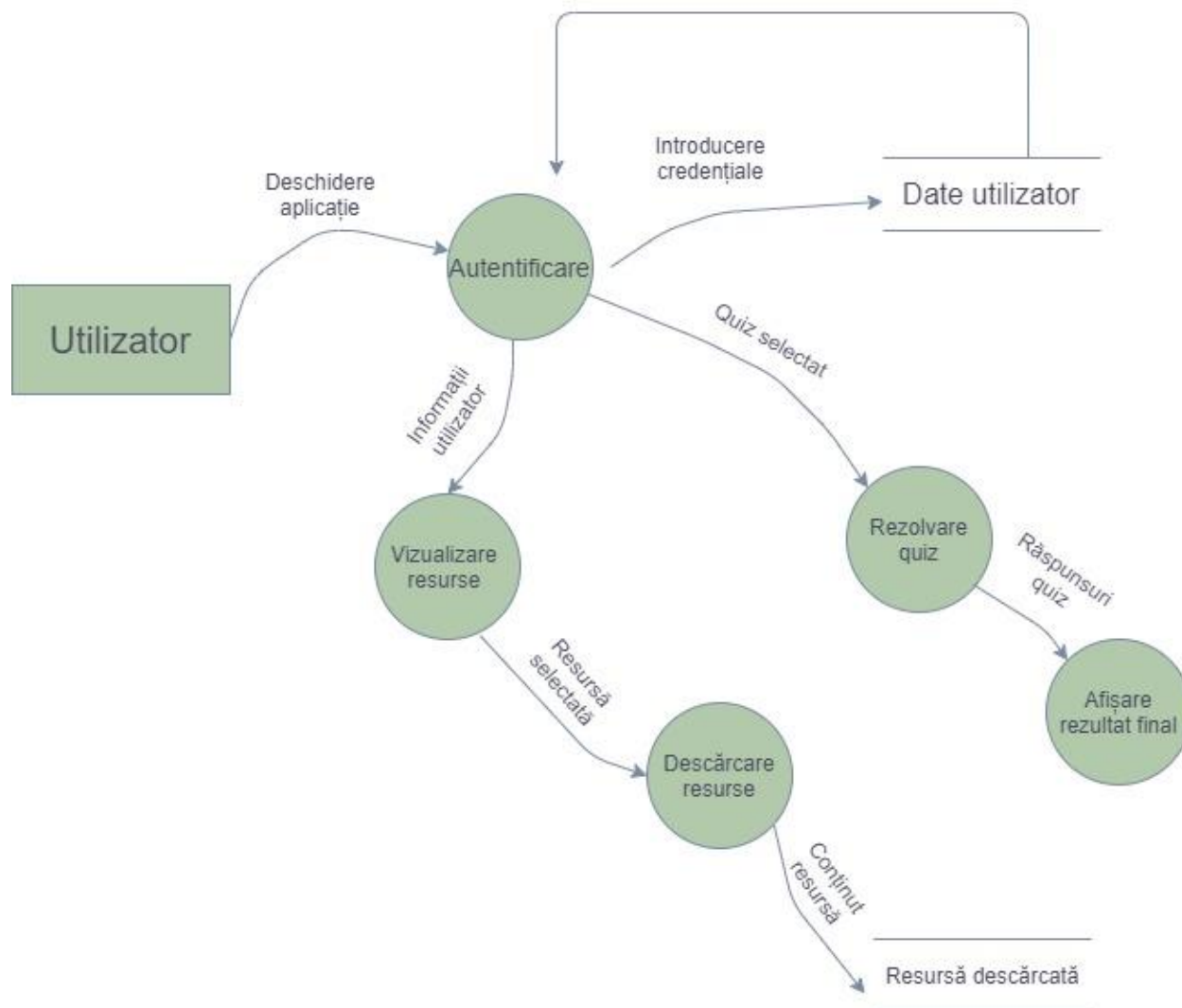


Figura 8 - Diagrama de Flux de Date, Adaptare după (Ciaca, 2021)

noastre am folosit arhitectura client-server, definită ca și concept pentru prima dată în anul 1980 (Yadav, 2009). Acest tip de sistem îndeplinește atât funcțiile clientului, cât și funcțiile serverului, prin promovarea transferului de informații dintre cele două părți (Sulyman, 2014). Arhitectura client-server este alcătuită din unul sau mai multe dispozitive conectate la un server central printr-o rețea sau conexiune la internet (Nagappan, Vayiravan, 2011). În cazul de față, clientul este reprezentat de aplicația care trimite cereri spre server, acesta din urmă le procesează și furnizează un răspuns. Gestionarea interfeței, procesarea inputurilor, transmiterea informației din baza de date sunt doar câteva dintre funcțiile îndeplinite de această arhitectură (Yadav, 2009). Figura 9 oferă o reprezentare grafică a noțiunilor discutate anterior și ne ajută să înțelegem mai bine modul de funcționare a sistemului din perspectiva acestei abordări.

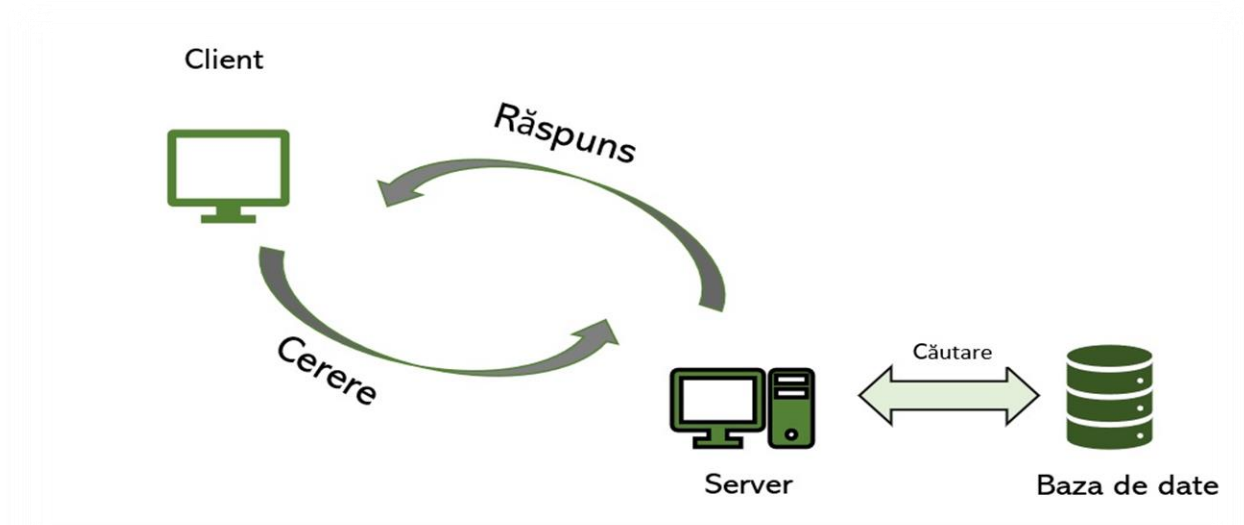


Figura 9 - Arhitectura Client – Server, Adaptare după (Khan, 2019)

Deducem din această figură modul de comunicare al clientului cu serverul și baza de date. Clientul sau aplicația, în cazul de față trimite o cerere către server, acesta face apel la informația stocată în bază de date cu ajutorul căreia furnizează un răspuns către client. Fiind o aplicație care gestionează și oferă resurse digitale s-a folosit acest tip de arhitectură care accentuează comunicarea clientului cu serverul pentru accesarea resurselor stocate într-o bază de date prin intermediul funcționalităților implementate. Un alt concept care vine în sprijinul sistemului client-server este Diagrama de Componente. Rolul acestei diagrame este de a arăta într-un mod structurat relațiile dintre componentele sistemului (Bell, 2004). Totodată oferă dezvoltatorilor un format natural pentru a începe modelarea unei soluții și permit verificarea existenței funcționalităților necesare prin regăsirea acestora printre componente. În plus, diagrama de componente este un instrument de comunicare util pentru diferite grupuri. Ea poate fi prezentată atât utilizatorilor cât și dezvoltatorilor (Bell, 2004).

RESURSE CREȘTINE DIGITALE PENTRU COPII

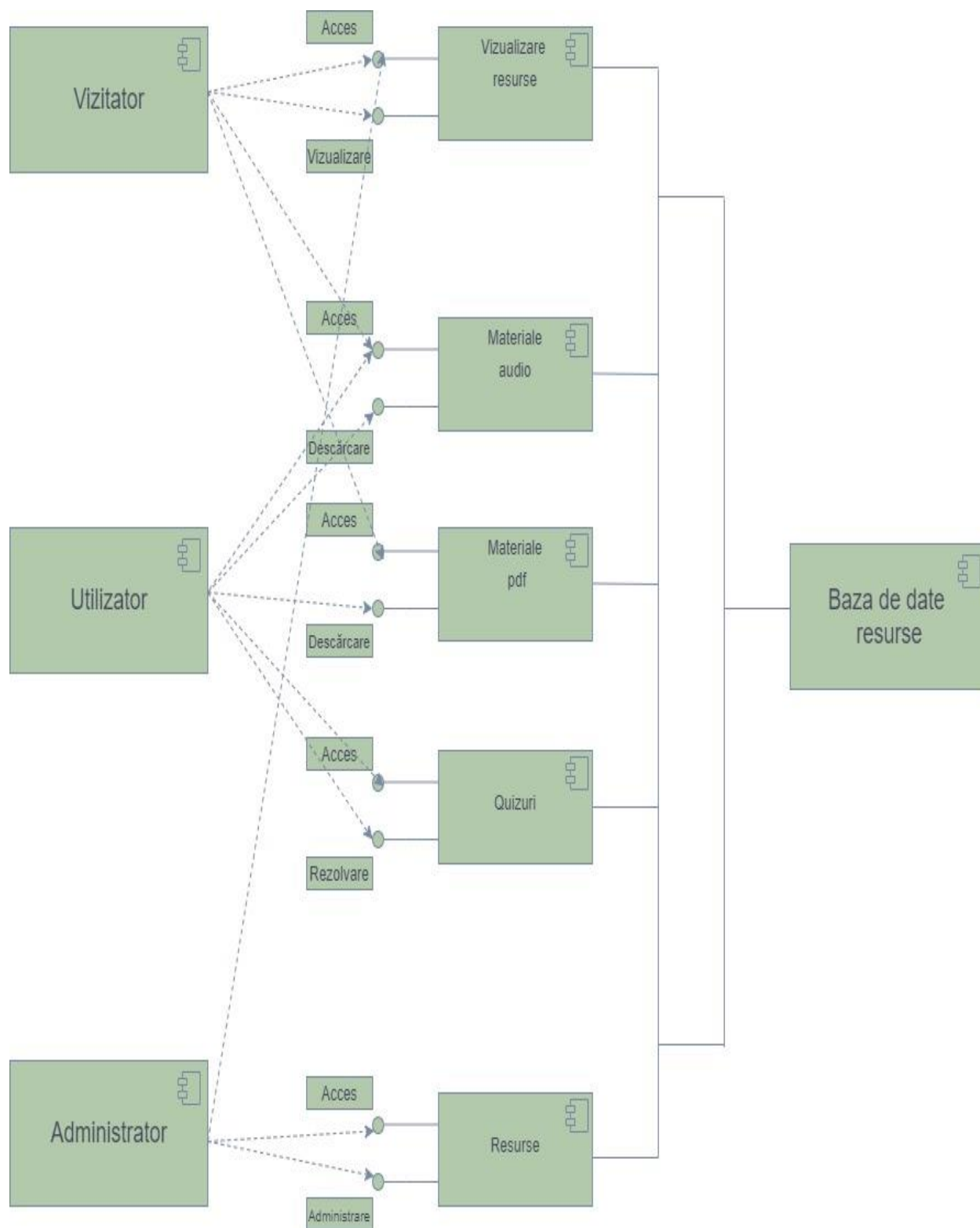


Figura 10 - Diagrama de componente, Adaptare după (Ciaca, 2021)

În cadrul acestui sistem avem trei componente care furnizează cele trei tipuri de utilizatori ai aplicației, fiecare având conexiuni specifice cu alte componente. Spre exemplu, componenta vizitator are acces la resursele aplicației în modul vizualizare. Continuăm cu exemplul componentei utilizator prin care se face referire la utilizatorul logat. Acesta poate accesa toate componentele resurselor, iar prin intermediul funcționalităților aferente poate interacționa cu acestea. Baza de date este ilustrată tot sub forma unei componente aflată în legătură cu toate celelalte prin furnizarea informațiilor suport.

4.2 Baza informațională

Baza informațională reprezintă totalitatea datelor prelucrate în sistem necesare pentru îndeplinirea scopului cu care acesta a fost implementat (Ciaca, 2021). Este una dintre cele mai importante componente ale sistemului, fără de care funcționarea acestuia ar fi inutilă. Importanța bazei informaționale se amplifică în aplicația noastră, a cărei rol principal este să ofere un suport informațional creștin destinat copiilor. Legat de structura bazei, a fost necesară abordarea cerințelor funcționale (Subcapitolul 2.3) pentru atingerea punctelor principale de interes pentru publicul utilizatorilor. Începem analizei bazei informaționale prin specificarea importanței datelor utilizatorilor. Pentru logarea în sistem fiecare user va introduce numele de utilizator și parola, regăsite alături de alte informații personale în baza de date. Un alt tip de utilizator este administratorul care logat cu contul aferent poate accesa și modifica resursele care alcătuiesc baza informațională.

Fiind vorba de o aplicație cu conținut creștin, informațiile au fost selectate din medii și surse specifice pentru armonizarea și alcătuirea unui întreg al resurselor. Am urmărit de asemenea ca informația să fie actualizată și în concordanță cu conținutul biblic pentru a asigura veridicitatea întregului suport prezentat. În ceea ce privește diversitatea datelor, s-a realizat prin expunerea materialelor în format text și audio ramificate în mai multe categorii. Am obținut astfel o sursă bogată de resurse concepută cu scopul de a îmbogăți cultura creștină a utilizatorilor. Toate aceste resurse prezentate mai sus au fost stocate într-o bază de date conectată la aplicația noastră. Vom trata în capitolul următor acest aspect privind organizarea datelor în tabele și legăturile dintre acestea.

4.3 Structura fizică a datelor

Vom începe acest capitol prin a defini conceptul bazei de date care este o colecție partajată de date între care se stabilesc relații logice, proiectată cu scopul de a satisface nevoile informaționale ale unei organizații (Connolly, 2001). După cum am menționat și în capitolul precedent, baza de date a sistemului reprezintă un punct foarte important în dezvoltarea acestuia, prin urmare necesită o structurare atentă și corectă pentru a-și putea îndeplini rolul cu succes. Proiectarea corectă a bazei de date impune o structurarea informației în tabele, a căror coloane reprezintă atributele iar fiecare rând conține o înregistrare. Construirea bazei de date specifică aplicației noastre s-a realizat în cadrul platformei phpMyAdmin, cu ajutorul serverului MySQL. În Figura 11 se pot observa tabelele care stochează informațiile necesare funcționării întregului sistem. Am optat pentru o bază de date relațională datorită facilităților oferite de relațiile dintre tabele prin reutilizarea informației de fiecare dată când este nevoie. Relațiile se stabilesc pe baza constrângerilor, care se definesc prin restricțiile privind legăturile dintre diferite entități (Sitar, 2020).

În cele ce urmează vom descrie fiecare tabel pentru a înțelege rolul acestuia în baza de date și relațiile care le are cu alte tabele.

Tabela Resurse stochează informațiile despre fiecare tip de resursă oferită utilizatorului, din această cauză este tabela cu cel mai mare număr de înregistrări. O resursă are titlu, autor, categorie, format, sursă sau conținut în funcție de categoria din care face parte. Conține un număr de trei chei străine, care fac legătura cu tabelele Categoriei, Format și Categoriei Quiz. Prima coloană reprezintă id-ul fiecărei resurse, care se autoincrementează, deținând astfel rolul de cheie primară.

Tabela User conține date despre utilizatorii aplicației, stocând id-ul unic al fiecărui user, numele, prenumele, numele utilizatorului, parola, număr de telefon, email și genul. Această tabelă se leagă de tabela Preferate unde sunt stocate resursele preferate al fiecărui utilizator.

Tabela Preferate menționată anterior stochează date despre resursele adăugate pe lista de preferințe al fiecărui utilizator, conținând două chei străine care fac legătura cu tabelele User și Resurse. Câmpurile acestei tabele conțin id-ul specific fiecărei resurse preferate din listă, id-ul resursei din tabela Resurse, id-ul utilizatorului, sursa, acolo unde este cazul și data adăugării în listă.

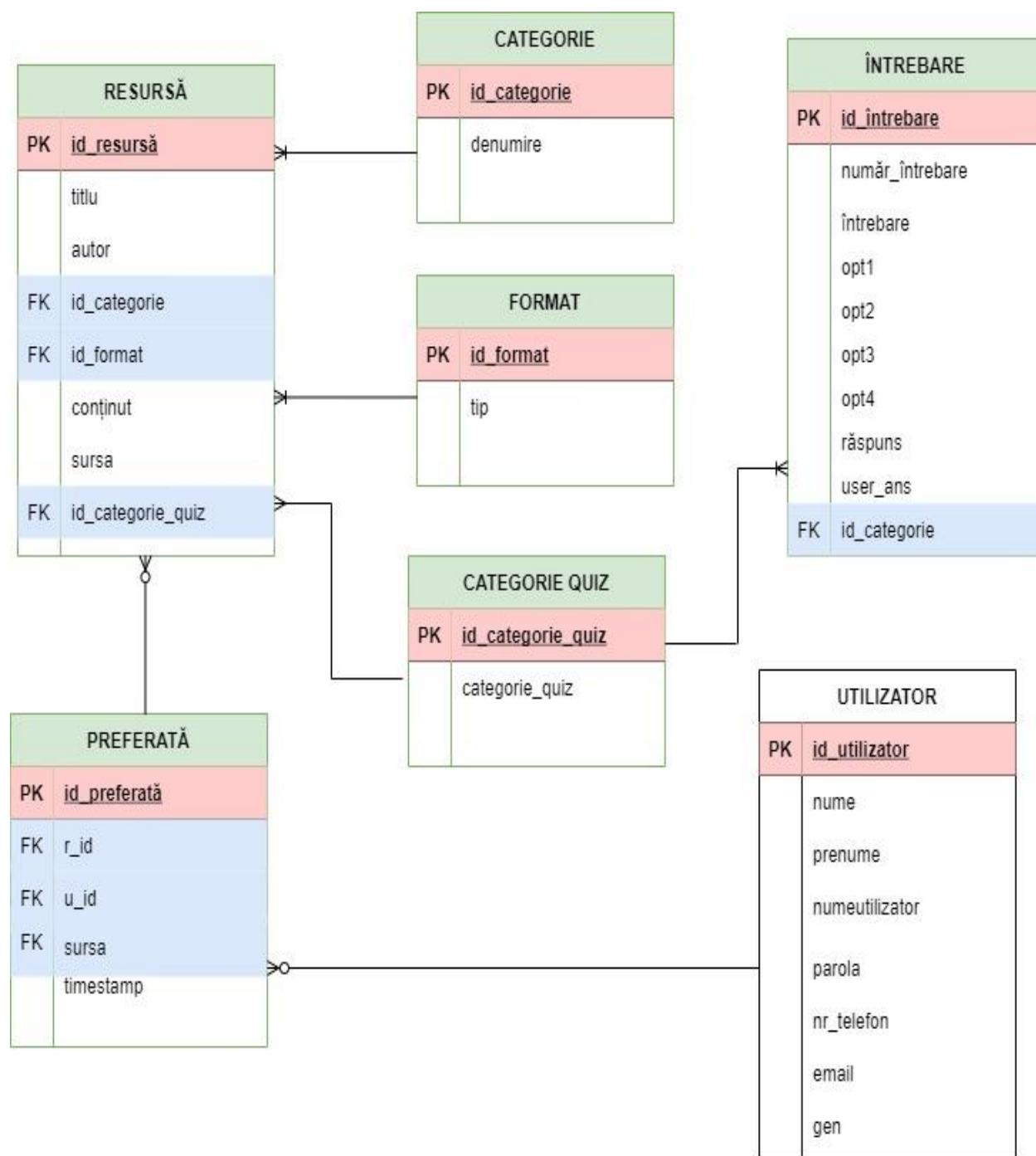


Figura 11 - Diagrama bazei de date, Adaptare după (Sitar, 2019)

Tabela Categorii conține id-ul și denumirea fiecărei categorii de resurse stocate în baza de date. Se leagă de tabela Resurse.

Tabela Format reprezintă tipul formatului în care sunt disponibile resursele. Deoarece în momentul actual dispunem doar de resurse text și audio, această tabelă conține doar aceste două înregistrări, legându-se de tabela Resurse.

Tabela Întrebări stochează întrebările aferente quizurilor puse la dispoziția utilizatorilor. Conține un număr de zece coloane în care regăsim, după cum urmează: id-ul unic al întrebării, reprezentat prin cheia primară, numărul întrebării, întrebarea propriu-zisă, cele patru variante de răspuns, răspunsul corect, răspunsul utilizatorului și categoria quizului din care face parte. Ultima coloană menționată este cheie străină prin care se creează legătura cu tabela Categorii Quiz.

În tabela Categorii Quiz se stochează cele două tipuri de quizuri din care fac parte întrebările. Astfel menționăm cele două înregistrări ale tabelii ca fiind Vechiul Testament și Noul Testament, mai concret cele două surse biblice pe baza cărora au fost formulate întrebările.

4.4 Procese și algoritmi

În cadrul acestui punct al lucrării vom exemplifica prin anumite funcționalități procedura și algoritmi folosiți în implementarea sistemului. Astfel, fiecare funcționalitate a aplicației a fost gândită prin prisma unui algoritm, unde s-a urmărit optimizarea procesului prin eficiență și resurse minime. Vom continua prin prezentarea algoritmului folosit în procedura de logare a utilizatorului.

Pas 1: Conectarea la baza de date.

Pas 2: Se selectează baza de date.

Pas 3: Se completează formularul de logare din pagina aferentă cu credențialele necesare, numele utilizatorului și parola.

Pas 4: Datele din formular sunt transmise spre validare.

Pas 5: Se verifică trimiterea datelor.

Pas 6: Se stochează în variabile datele preluate din formular.

Pas 7: Se stochează într-o variabilă interogarea în cadrul căreia se extrag credențialele utilizatorului.

Pas 8: Se execută interogarea de mai sus în interiorul bazei de date.

Pas 9: Se verifică dacă rezultatul interogării este cel așteptat.

Pas 10: Obținerea unui rezultat diferit de 1 returnează un mesaj de eroare.

Pas 11: Altfel se continuă prin inițializarea unei variabilei cu numele utilizatorului logat și trimiterea în pagina Home.

Pas 12: Se verifică dacă contul aparține administratorului și se face trimiterea în pagina corespunzătoare.

După cum se poate observa în pseudocodul prezentat, autentificarea în sistem se face prin completarea de către utilizator a unui formular care conține câmpurile pentru numele de utilizator și parolă. Procesul se continuă prin verificarea corespondenței dintre datele transmise și conținutul bazei de date din tabela cu datele utilizatorilor. Succesul acestei proceduri rezultă din trimiterea în pagina de Home și apariția în bara de meniu din partea de sus, un buton pe care este afișat numele utilizatorului logat în aplicație (Figura 12).



Figura 12 - Bara de meniu

Continuăm exemplificarea prin analiza unei operații CRUD, alegând-o pe cea de modificarea unei resurse din aplicație.

Pas 1: Conectarea la baza de date.

Pas 2: Selectarea bazei de date.

Pas 3: Se execută o interogare prin care se selectează poeziile din baza de date;

Pas 4: Se verifică rezultatul interogării și se afișează sub forma unui tabel lista cu titlurile, autorii și butonul pentru modificare aferente fiecărei poezii;

Pas 5: Se apasă butonul pentru poezia care se dorește a fi editată și se face trimiterea spre pagina unde se realizează procesul de modificare;

Pas 6: Se verifică dacă câmpurile formularului de modificare au fost completate;

Pas 7: Se execută interogarea cu datele preluate;

Pas 8: Se afișează un mesaj de confirmare;

Analog, se prezintă algoritmi pentru toate celelalte funcționalități din cadrul aplicației.

4.5 Tehnologii specifice

Aplicația Resurse creștine digitale pentru copii a fost implementată cu ajutorul mai multor tehnologii specifice, cu diferite roluri. Fiecare limbaj de programare a avut un rol bine definit în realizarea întregului sistem. Pentru interfață s-a folosit HTML, CSS, JavaScript și librăria Bootstrap, partea de background fiind implementată prin PHP și MySQL. În cele ce urmează vom dezbate pe rând fiecare tehnologie și implicarea ei în construirea aplicației.

HTML (Hypertext Markup Language) este un limbaj folosit pentru descrierea structurii unei pagini web, având definite un set de structuri comune, cum ar fi: titluri, paragrafe, liste, tabele (Colburn, 2015). Fiind un limbaj de marcare, HTML folosește taguri speciale pentru a identifica diferite elemente din conținutul paginii, care produc diferite rezultate în cadrul browserului folosit. Conținutul unei pagini HTML este format din textul paginii și tagurile folosite în structură, formatare și crearea legăturii paginii respective cu alte pagini web (Kyrnin, 2015). În aplicația noastră, cu ajutorul limbajului HTML am creat paginile regăsite în structura meniului, tabelele și listele pentru organizarea datelor, formularul de contact și alte elemente care țin de modalitatea de expunere a resurselor. Formatarea elementelor HTML s-a realizat prin folosirea limbajului CSS (Cascading Style Sheets). Într-o manieră simplă de-l defini, CSS este modul prin care se separă structura documentului de aspectul său (Meyer, 2006). Acest standard de formatare prezintă numeroase avantaje: permite o stilizare aprofundată a elementelor HTML, este competent în modificarea a numeroase pagini printr-un singur document, prezintă o sintaxă simplă, ușor de înțeles (Meyer, 2006). Am ales stilizarea externă prin utilizarea fișierelor separate care pot fi reutilizate într-un alt context, acesta reprezentând și un avantaj al acestui limbaj (Knap, 2017).

Alături de cele două limbaje prezentate anterior și folosite în dezvoltarea interfeței, se alătură limbajul de scripting JavaScript cu rol în creșterea interactivității paginilor web (Haverbeke, 2018). Se poate afirma despre acest limbaj că programează comportamentul paginilor web, prin caracteristica dinamică pe care o oferă. Un exemplu de utilizare în codul aplicației noastre este în derularea imaginilor aferente caruselului afișat pe prima pagină a aplicației. Cu ajutorul unei funcții putem schimba imaginea afișată prin apăsarea săgeților cu rol de next și prev aflate pe margine. De asemenea, în interiorul codului regăsim și alte exemple de utilizare a funcțiilor JavaScript.

Puntea dintre client și server este trecută cu ajutorul limbajelor PHP și MySQL. Limbajul PHP este un limbaj de scripting server-side utilizat pentru crearea paginilor dinamice (Stephens, 2005). Este unul dintre cele mai utilizate limbaje datorită avantajelor care le oferă printr-o sintaxă ușor de scris, open-source, caracter portabil. (Vaswani, 2009). Scripturile PHP sunt executate pe server realizându-se astfel conexiunea cu baza de date. Extragerea datelor din baza informațională se face prin intermediul interogărilor sistemului de gestiunea a bazelor de date MySQL. Construirea și administrarea bazei de date se realizează prin phpMyAdmin, un instrument PHP destinat administrării bazelor de date prin intermediul unui browser web.

În capitolul următor vom aborda aspecte legate de implementarea sistemului prin prezentarea funcționalităților relevante cu explicațiile aferente și capturi de ecran pentru vizualizarea rezultatelor.

5. Implementarea sistemului

Acest capitol abordează un subiect foarte important din componența acestei lucrări. Prin intermediul lui putem observa rodul muncii de până în acest moment și totodată vom discuta aspectele importante ale aplicației atât din perspectiva codului sursă, cât și a rezultatelor vizibile în interfața grafică. Vom începe cu descrierea funcționalității de autentificare în aplicație.

Autentificarea în sistem se face prin completarea de către utilizator a unui formular a cărui câmpuri corespund coloanelor tabelului utilizatori din baza de date, deoarece în mod automat înregistrarea unui nou utilizator presupune inserarea unei noi înregistrări în baza de date.

Preluarea datelor din formular s-a realizat cu ajutorul variabilei superglobală \$_POST, folosită în PHP pentru colectarea datelor (Stanca, 2020). Următorul pas a constat în stocarea datelor preluate de la utilizator în variabile cu ajutorul funcției mysqli_real_escape_string() (Stephens, 2005), care înlătură posibilele caractere speciale din datele introduse de utilizator. Procedura continuă prin inserarea acestor variabilelor în baza de date și se încheie cu direcționarea în pagina de logare pentru a confirma funcționarea contului creat.

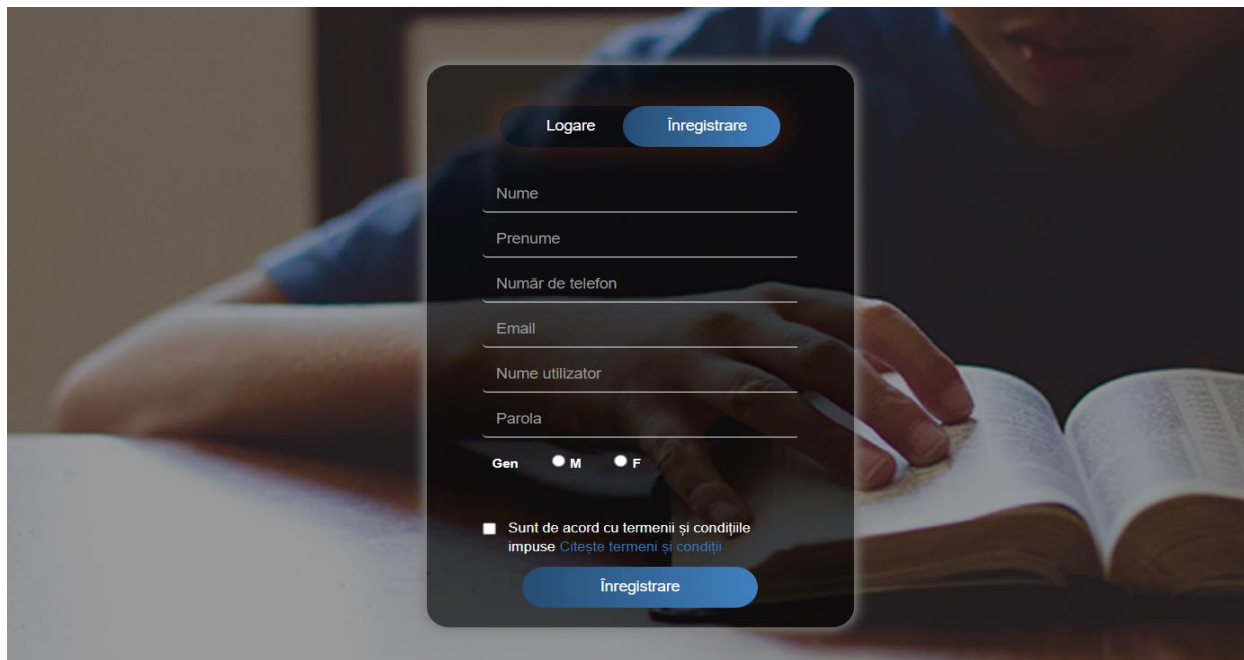
```

16
17 $mysqli=new mysqli($servername, $username, $password, $db);
18
19 if ($mysqli->connect_error) {
20     die("Connection failed: " . $con->connect_error);
21 }
22
23 $nume = mysqli_real_escape_string($mysqli, $_POST['nume']);
24 $prenume = mysqli_real_escape_string($mysqli, $_POST['prenume']);
25 $numeutilizator = mysqli_real_escape_string($mysqli, $_POST['numeutilizator']);
26 $parola = mysqli_real_escape_string($mysqli, $_POST['parola']);
27 $nr_telefon = mysqli_real_escape_string($mysqli, $_POST['nr_telefon']);
28 $email = mysqli_real_escape_string($mysqli, $_POST['email']);
29 $gen = mysqli_real_escape_string($mysqli, $_POST['gen']);
30
31 $s = "SELECT * FROM user WHERE nume='$nume' LIMIT 1";
32
33 $result = mysqli_query($mysqli, $s);
34
35 $num=mysqli_num_rows($result);
36
37 if($num == 1){
38     header('location:newlogin.php?error');
39 }
40 }else{
41     $reg = "INSERT INTO user (nume , prenume, numeutilizator , parola , nr_telefon , email , gen) VALUES ('$nume' , '$prenume' , '$numeutilizator' , '$parola' , '$nr_telefon' , '$email' , '$gen')";
42     mysqli_query($mysqli , $reg);
43     header('location:newlogin.php?success');
44 }
45 mysqli_close($mysqli);
46 ?>

```

Figura 13 - Funcția de înregistrare

Formularul de înregistrare este prezentat în Figura 14, unde se poate observa existența unui câmp aferent fiecărei coloane din baza de date care stochează datele utilizatorilor. În urma completării acestui formular se va crea contul de utilizator pe baza căruia va fi posibilă logarea.



The registration form is displayed over a background image of a child reading a book. The form itself is a dark, semi-transparent rectangle. At the top, it has two buttons: 'Logare' (Login) and 'Înregistrare' (Register), with 'Înregistrare' being the active one. Below these are input fields for 'Nume' (Name), 'Prenume' (First Name), 'Număr de telefon' (Phone Number), 'Email', 'Nume utilizator' (Username), and 'Parola' (Password). There are also radio buttons for 'Gen' (Gender) with options 'M' (Male) and 'F' (Female). A checkbox is present for 'Sunt de acord cu termenii și condițiile impuse' (I agree with the terms and conditions imposed), with a link 'Citește termenii și condițiile' (Read terms and conditions). At the bottom of the form is another 'Înregistrare' button.

Figura 14 - Formularul de înregistrare

Scopul principal al aplicației este îndeplinit prin punerea la dispoziția utilizatorului resurse din diferite categorii și diferite formate. Utilizatorul logat are accesul permis la toate resursele aplicației, în timp ce un vizitator sau denumit astfel utilizatorul nelogat este restricționat în accesarea anumitor pagini. Unele resurse pot fi descărcate, oferind utilizatorului posibilitatea de a beneficia de resurse și atunci când utilizatorul nu poate accesa aplicația. Resursele regăsite în aplicație se regăsesc în meniul inclus în aproape toate paginile în partea stângă. Putem observa diversitatea resurselor, cât și celalalte opțiuni de utilizare a aplicației în Figura 16. Meniul a fost implementat cu ajutorul tag-ului care definește o listă neordonată, iar fiecare element din listă este însoțit de tag-ul (Colburn, 2015). Atunci când utilizatorul selectează printr-un click categoria dorită va fi redirectionat spre pagina ce conține resursele aparținătoare categoriei respective. Alături de aceste butoane regăsim o bară de căutare care în funcție de valoarea introdusă va returna resursa a cărei titlu conține cuvântul introdus. Implementarea codului sursă a formularului prin care se preia valoarea introdusă și se transmite prin metoda POST paginii de generare a rezultatului, se poate consulta în Figura 15.


```

18
19 <div class="search-container">
20   <form action="searchbar.php" method="POST">
21     <input type="text" placeholder="Caută o resursă.." name="search">
22     <button type="submit" name="submit-search"><i class="fa fa-search"></i></button>
23   </form>
24 </div>

```

Figura 15 - Formular bară de căutare

Pe prima pagină regăsim, de asemenea, un carusel cu imagini reprezentative pentru fiecare tip de resursă. În meniul din partea de sus se observă câteva butoane care fac legătura cu câteva pagini a căror conținut este indicat de textul afișat pe buton. Butonul cu numele utilizatorului logat aduce utilizatorul în pagina principală atunci când acesta navighează în celelalte pagini. Butonul de preferințe conduce spre pagina unde sunt salvate resursele preferate a utilizatorului logat. Următorul buton care se poate observa în figură trimite în pagina de contact, unde este disponibil un formular prin intermediul căruia utilizatorul poate trimite un mesaj pe email administratorului aplicației. Apăsarea ultimului buton reprezintă delogarea din sistem și direcționarea spre pagina de autentificare.

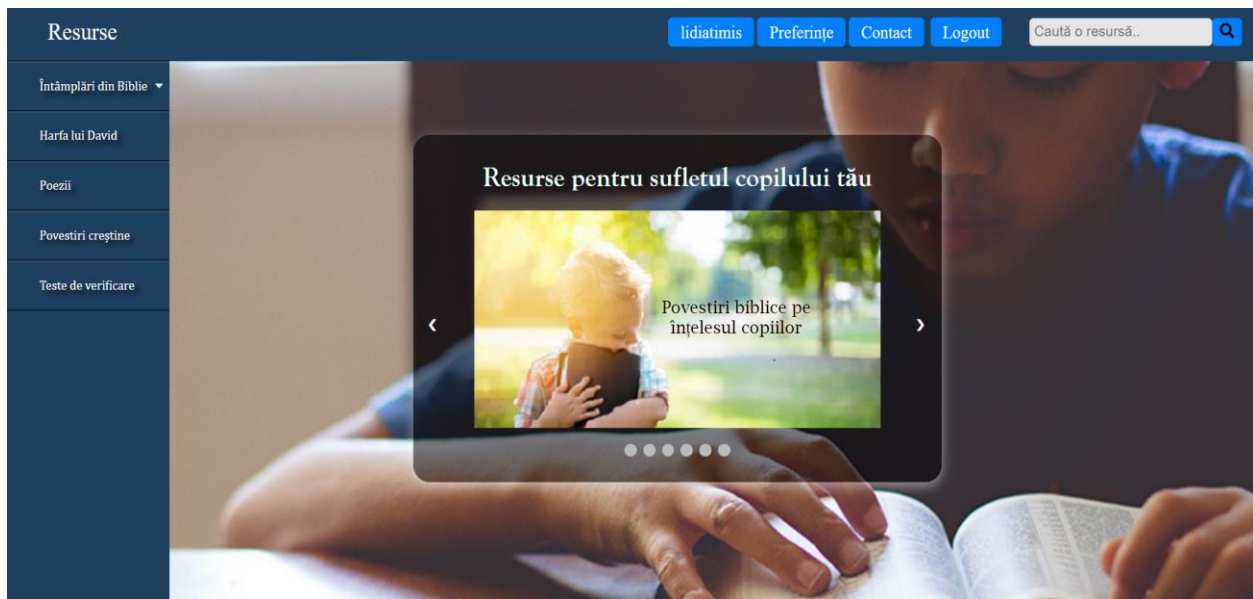


Figura 16 - Pagina Home

Una dintre categoriile de resurse din aplicație este cea a poeziilor. Baza de date stochează un număr de douăzeci de poezii afișate într-o listă care se împarte pe două pagini. Atunci când utilizatorul alege o poezie din listă printr-un click pe titlul acesteia este direcționat într-o pagină generică unde va fi afișat titlul, autorul și conținutul poeziei. Generarea acestei pagini s-a realizat pe baza id-ului din baza de date a fiecărei poezii transmis prin metoda GET (Stanca, 2020).

```
while($row= mysqli_fetch_array($resultSet)) {  
    echo"<a id='link' href='poezie.php?id={$row['id_resursa']}'>{$row['titlu']}</a><br></n>";  
    echo '<br>'; ?>
```

Pagina poezie.php preia id-ul transmis în interiorul link-ului, îl salvează într-o variabilă \$id pe baza căreia se interoghează baza de date și se afișează în pagină datele corespunzătoare id-ului primit. În această pagină regăsim și un buton care permite adăugarea poeziei în lista de preferate, oferind utilizatorului posibilitatea de a economisi timpul de căutare a unei resurse apreciate prin regăsirea acestei în pagina cu resurse preferate. Aceste aspecte se pot observa în Figura 17.



Figura 17 - Pagina generică a unei poezii

O altă categorie importantă de resurse sunt cântecele în format audio. În această categorie conținutul este în conformitate cu tendințele actuale și apartenența în sfera creștină, alegând un număr de trei interpreți preferați ai publicului, dar am creat și o secțiune cu câteva cântece în limba engleză pentru iubitorii acestei limbi. Fiecare din aceste subcategorii le-am afișat în pagină sub forma unui buton care conține link-ul spre pagina cu cântecele aparținătoare interpretului respectiv. Modul în care am afișat aceste butoane în codul sursă se poate vedea în Figura 18.

[illegible]

Figura 18 - Secțiunea de cântece

În capitolul următor vom vorbi despre testarea sistemului, unde vom aborda testarea white box și black box, prin exemplificarea în cadrul sistemului nostru.

6. Testarea sistemului

Acest capitol este un instrument de ajutor în verificarea funcționării corecte a sistemului dezvoltat până acum. Etapa de testare este foarte importantă deoarece are ca scop asigurarea unei implementări reușite și garantarea că produsul software obținut satisface nevoile clienților (Diaconu, 2003). Vom prezenta această noțiune din perspectiva mai multor abordări. Interesul nostru principal este testarea funcționalității sistemului care se poate realiza în mod static sau dinamic. Testarea statică este forma de testare care nu implică executarea programului, pe când testarea dinamică implică programul într-un mod direct prin executarea sa (Jecan, 2020).

Ramificarea continuă pe ramura testării dinamice care la rândul ei poate fi white box (Jecan, 2020) – testerul are acces la codul sursă sau black box (Jecan, 2020), numită și testarea comportamentală în care se testează comportamentul programului fără accesul la codul sursă (Jecan, 2020). În cele ce urmează vom testa aplicația noastră prin cele două metode: testarea dinamică black box și white box. După cum am menționat și în paragraful anterior testarea dinamică black box presupune ieșirile programului la aplicarea unor stimuli sau intrări. Vom exemplifica această abordare în interiorul procesului de logare, prin introducerea unor date care nu se găsesc în baza de date. De exemplu vom încerca să ne logăm cu un nume de utilizator incomplet, dar cu o parolă corectă. Rezultatul îl vom analiza în Figura 19.

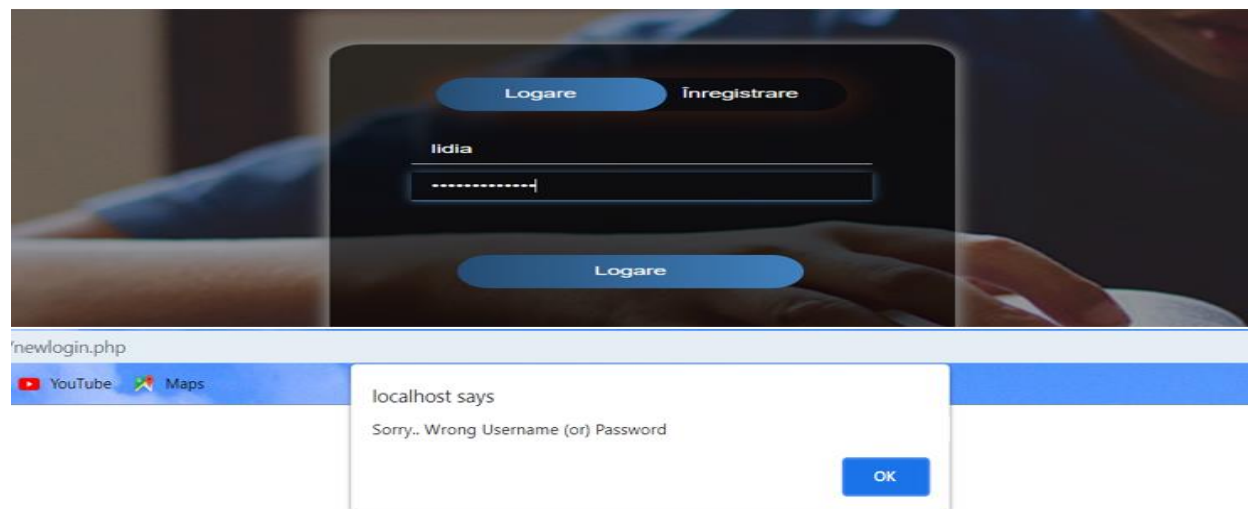


Figura 19 - Caz de testare 1

RESURSE CREȘTINE DIGITALE PENTRU COPII

Figura următoare prezintă acest caz de test prezentat sub forma unui tabel, în care sunt prezentate specificațiile testării.

Număr raport	01	
Data	18.06.2021	
Denumire aplicație	Resurse creștine digitale pentru copii	
Hardware	HP 15s-fq2015nq, FHD, Procesor Intel® Core i5-1135G7, 8GB DDR4, 512GB SSD, Intel Iris Xe, Free DOS	
Sistem operare	Windows 10 Education 20H2	
Denumire eroare	Logare nereușită	
Tip eroare	Permanentă	
Gravitate eroare	Majoră	
	Medie	X
	Minoră	
Status eroare	Rezolvat	X
	Nerezolvat	
Descriere eroare	Nu se poate realiza logarea în aplicație, deoarece numele utilizatorului este greșit.	
Origine eroare	Formular de logare	
Autor depistare eroare	Timiș Lidia-Ancuța	
Autor corectare eroare	Timiș Lidia-Ancuța	

Tabel 2 - Caz de testare 1, Adaptare după (Jecan, 2020)

Un alt caz de testare îl vom face prin metoda dinamică white box, unde vom avea acces la codul sursă. Dorim să afișăm titlurile povestirilor în format pdf prin interogarea bazei de date, aplicând clauza WHERE cu condițiile necesare pentru selectarea conținutului specific. După scrierea codului testăm rezultatul în aplicație, însă se observă o eroare comună pentru fiecare rând, Figura 21.

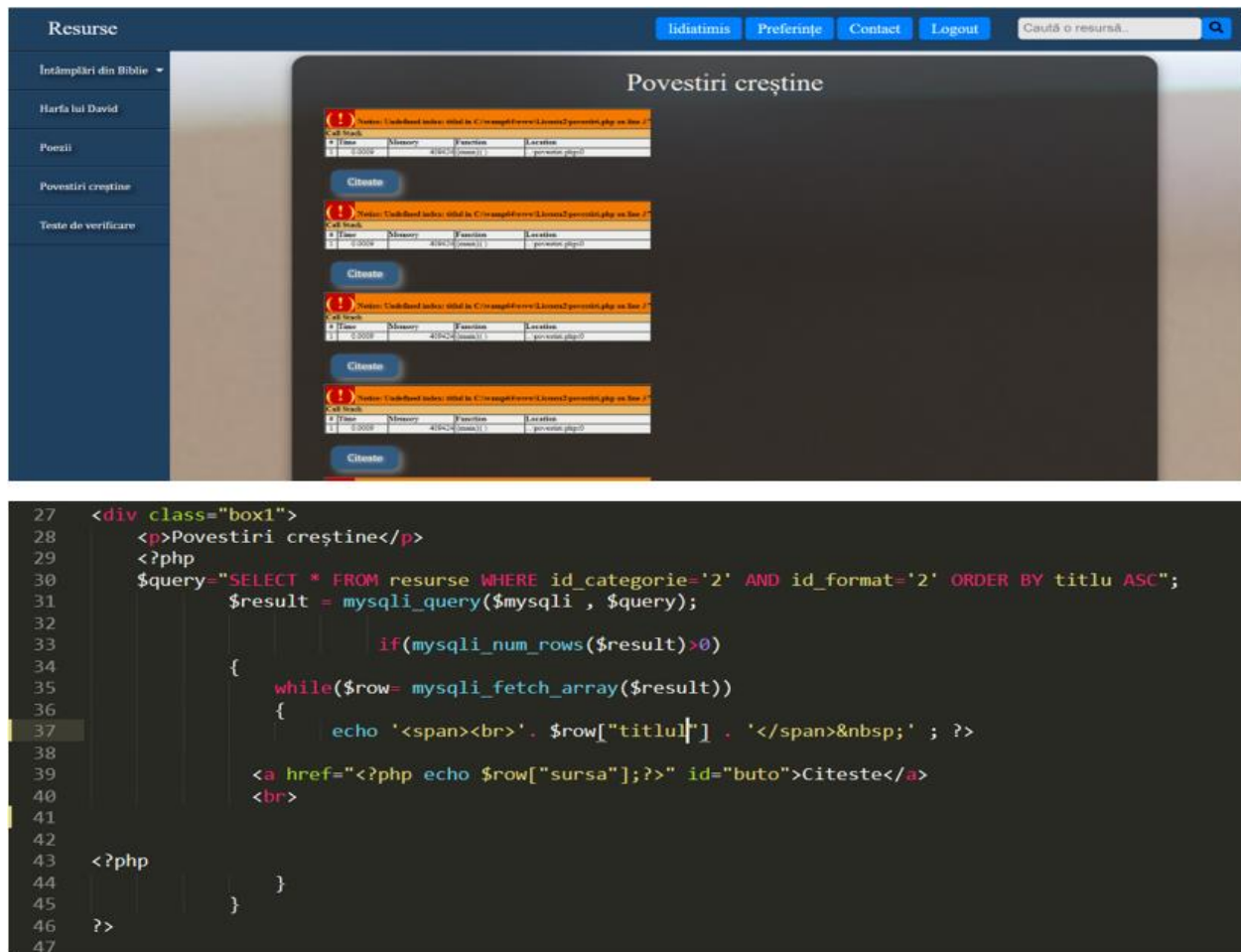


Figura 20 - Eroare – Caz testare 2

Eroarea definită transmite mesajul că indexul *titlu* nu este definit, ceea ce sugerează o necorespondență între numele coloanei respective din baza de date și numele scris în codul sursă. Verificând numele din baza de date vom observa că acesta este titlu, nu *titlu*, cum este scris în cod. Corectarea acestei greșeli conduce la dispariția erorii și afișarea corectă în pagină a conținutului dorit, fapt ilustrat în Figura 21.

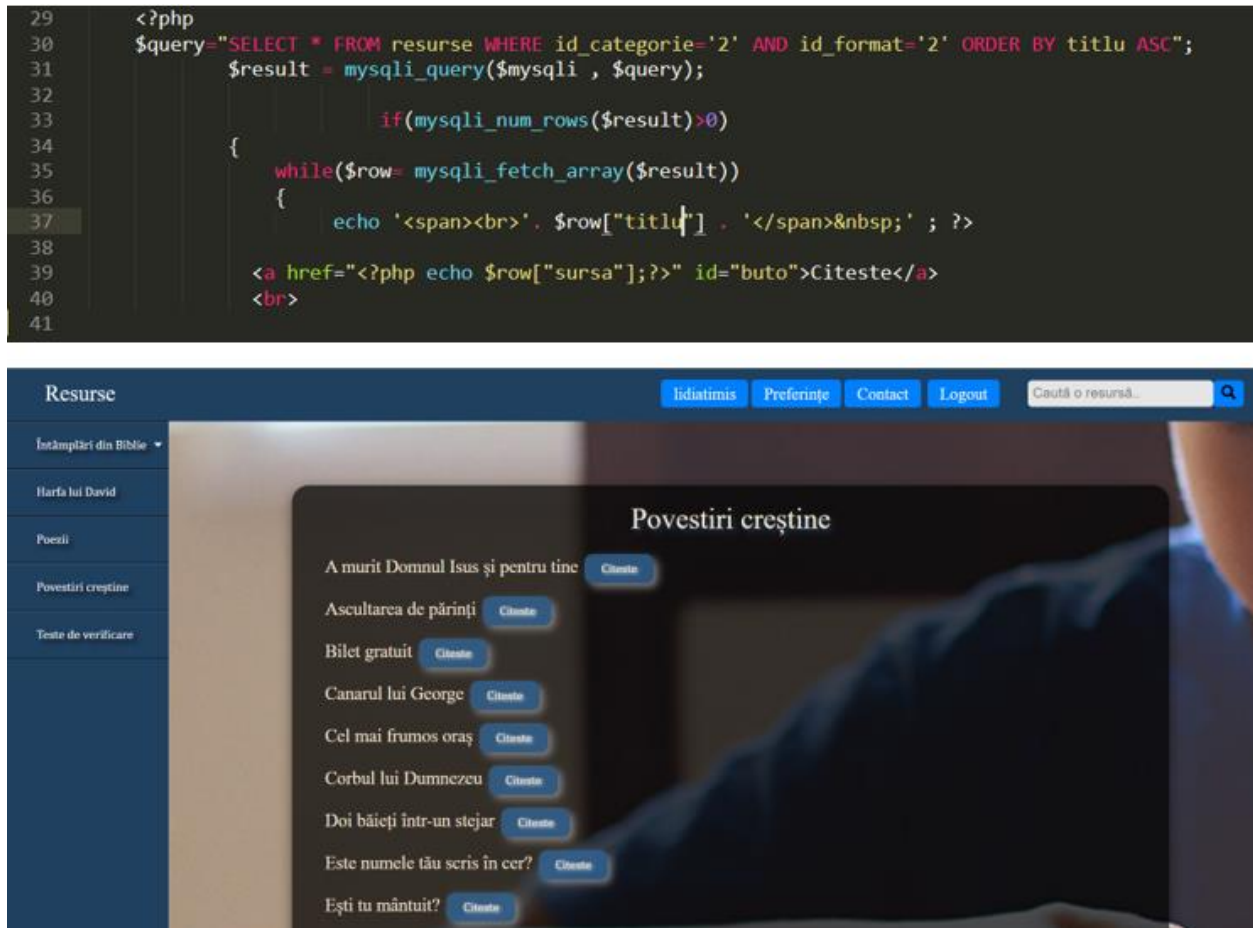


Figura 21 - Rezolvare Eroare – Caz testare 2

Specificațiile celui de-al doilea caz de testare se observă în Tabelul 2, unde regăsim detalii specifice despre eroarea întâmpinată.

Cazurile de testare pot continua pe tot parcursul implementării aplicației. Deoarece sistemul nostru are un grad de complexitate scăzut și nu se adresează unui public extins, rolul testerului a fost preluat de programator prin testarea funcționalităților la fiecare pas al implementării codului sursă. Această modalitate a asigurat construirea treptată a unui produs software reușit care să satisfacă utilizatorul prin funcționarea completă și corectă în momentul livrării produsului.

Vom încheia această lucrare cu secțiunea următoare de concluzii, unde vom prezenta concluziile desprinse din proiectarea sistemului și direcțiile viitoare de dezvoltare.

RESURSE CREȘTINE DIGITALE PENTRU COPII

Număr raport	02	
Data	18.06.2021	
Denumire aplicație	Resurse creștine digitale pentru copii	
Hardware	HP 15s-fq2015nq, FHD, Procesor Intel® Core i5-1135G7, 8GB DDR4, 512GB SSD, Intel Iris Xe, Free DOS	
Sistem operare	Windows 10 Education 20H2	
Denumire eroare	Interogare nereușită	
Tip eroare	Permanentă	
Gravitate eroare	Majoră	
	Medie	
	Minoră	X
Status eroare	Rezolvat	X
	Nerezolvat	
Descriere eroare	Nu se pot afișa titlurile povestirilor pdf în interfață din cauză că în interogarea bazei de date este o greșeală de scriere.	
Origine eroare	Interogarea SELECT	
Autor depistare eroare	Timiș Lidia-Ancuța	
Autor corectare eroare	Timiș Lidia-Ancuța	

Tabel 3 - Caz de testare 2, Adaptare după (Jecan, 2020)

Concluzii

Tema acestei lucrări constă în realizarea unei aplicații web care oferă un suport de resurse creștine digitale pentru copii. Scopul acestei aplicații este de a veni în sprijinul copiilor dornici de a se familiariza cu mediul creștin bazat pe cultura biblică, prin accesarea mai multor categorii de resurse în diferite formate. Materialele puse la dispoziție au un conținut specific temei, au fost atent selectate și acoperă următoarele categorii: povestiri audio relatate pe baza conținutului biblic din Noul și Vechiul Testament, un set de cântece creștine în format audio intitulat Harfa lui David, poezii în format text scrise de diferiți autori, povestiri pdf și o secțiune care conține două quizuri cu întrebări de cultură generală biblică. Toate acestea sunt la dispoziția utilizatorului prin intermediul unei interfețe grafice prietenoase, ușor de utilizat și atractivă pentru copii. De-a lungul implementării s-a urmărit interacțiunea sistemului cu utilizatorul prin crearea unor funcționalități care îl implică activ pe acesta. Amintim dintre acestea posibilitatea de a descărca anumite resurse, crearea unei liste de resurse preferate, rezolvarea quiz-urilor. În același timp sunt privați de aceste drepturi vizitatorii site-ului sau utilizatorii nelogați. Restricția de acces se manifestă prin direcționarea către pagina de logare atunci când se încearcă accesarea unor zone pentru care este necesară autentificarea. Un alt tip de utilizator este administratorul care prin logarea cu un cont specific intră într-o zonă din care poate efectua operațiile CRUD asupra resurselor existente. Realizarea tuturor aspectelor menționate mai sus a fost posibilă cu ajutorul limbajelor HTML, CSS, JavaScript pentru partea de interfață și PHP, MySQL folosite pentru implementarea părții de backend, incluzând partea de server și baza de date. În dezvoltarea acestei aplicații am întâmpinat diverse provocări care au constituit un ajutor în dobândirea experienței de lucru în domeniul programării procedurale folosind ca și limbaj PHP. Motivația avută la bază este definită prin crearea unui cerc de siguranță în mediul online în interiorul căruia se regăsesc resurse care ajută la dezvoltarea caracterului creștin a unui copil.

Direcțiile viitoare de dezvoltare prezintă o varietate îmbunătățită a aplicației care să ofere mai multe funcționalități și să sporească interacțiunea cu utilizatorul. De exemplu, ne propunem să dezvoltăm secțiunea de quizuri prin adăugarea mai multor seturi de întrebări și salvarea rezultatelor într-un istoric pe care utilizatorul să-l poată vedea atunci când este logat în aplicație. De asemenea, ne dorim să extindem diversitatea resurselor prin adăugarea de materiale video, cum ar fi desene animate cu conținut creștin, filmulețe educative sau cântece cu videoclip atașat.

Putem adăuga la cele de mai sus și implementarea unei pagini care să conțină cărți creștine pentru copii în format pdf în sprijinul iubitorilor de literatură. Vom încheia această lucrare menționând că dezvoltarea unui sistem informatic destinat unui public format din cea mai sensibilă și specială categorie de vârstă a fost o muncă susținută, dar făcută cu plăcere, obținând un rezultat satisfăcător. Consider că această aplicație își va îndeplini scopul și va fi o sursă de hrană spirituală într-un context tehnologic actual care nu promovează conservarea principiilor creștine, cu precădere în rândul generației viitoare. De aceea, “sacrificându-ne astăzi, copiii noștri vor avea mâine un mai bine” (Kalam, 2000).

Bibliografie

- Ajzele, B., Bierer, D. și Hussain, A. (2016), *PHP 7: Real World Application Development*, Editura Packt Publishing, Birmingham.
- Babin, L., Good, N., Kromann, M. și Stephens, J. (2005), *PHP 5 Recipes: A Problem-Solution Approach*, Editura Springer-Verlag, New York.
- Buchmann, R. și Jecan, S. (2020), *Testarea produselor software*, Note curs.
- Ciaca Monica (2021), *Analiza și proiectarea sistemelor informatice*, Note curs.
- Collison Simon (2006), *Beginning CSS Web Development: From Novice to Professional*.
- Sitar-Tăut Dan-Andrei (2020), *Baze de date în economie*, Note curs.
- Stanca Liana (2020), *Proiectarea și realizarea site-urilor și Portalurilor WEB*, Note curs.
- Stepan Aurel (2009), *Sisteme informatice și Inteligența artificială în economie*, Note curs.
- Schwaber, K. și Sutherland, J. (2017), *Ghidul Scrum*.
- Yadav, S.C. și Sing S.K. (2009), *An Introduction in Client/Server Computing*.
Editura New Age International Publishers, New Delhi.
- International Institute of Business Analysis (2006), *Guide to the Business Analysis Body of Knowledge*.
- Client-Server Model, <https://www.geeksforgeeks.org/client-server-model/>, Accesat la data de 11.06.2021.
- PHP Tutorial, <https://www.w3schools.com/php/default.asp>, Accesat la data de 10.06.2021.

Anexe

Anexa 1. **Tabel 4-** Scenariu de utilizare 2, Adaptare după (Ciaca, 2021)

Secțiune	Conținut
Identificator	A.R.C
Nume	Aplicație web care gestionează resurse creștine pentru copii
Autor	Timiș Lidia-Ancuța
Versiunea	V.0.1
Prioritate	Medie
Severitate	Medie
Responsabil	Timiș Lidia-Ancuța
Scurtă descriere	Administratorul se loghează în sistem cu un cont special de admin, selectează resursă asupra căruia se va exercita dreptul de administrare și se va selecta opțiunea de ștergere.
Tip de scenariu	Scenariu de interacțiune (administrator-sistem)
Obiectiv	Ștergerea de către administrator a unei resurse din aplicație.
Actori	Administrator și sistem
Precondiție	Persoana care dorește să șteargă resursa de pe site, trebuie să fie logată cu un cont de administrator care să-i ofere dreptul de operații CRUD asupra materialelor.
Postcondiție	Resursa ștearsă nu mai apare atât în interfață, cât nici în baza de date a sistemului.
Pașii scenariului	<ol style="list-style-type: none"> 1. Administratorul deschide aplicația web și se loghează cu contul aferent. 2. Se caută resursa care se dorește a fi ștearsă. 3. Apăsând butonul de ștergere, administratorul șterge resursa din componența sistemului. 4. În momentul următor, resursa nu va mai fi vizibilă și existentă pentru niciun tip de utilizator al aplicației.
Calitate	Aplicația trebuie să aibă partea de administrare din interfață.
Relații cu alte scenarii	Logarea administratorului în sistem

RESURSE CREȘTINE DIGITALE PENTRU COPII

Anexa 2. Chestionar privind utilitatea unei aplicații web care oferă resurse creștine pentru copii

<https://forms.gle/JnEToPvLwnVGuivU7>