

UNIVERSITATEA DIN BUCUREȘTI
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ
SPECIALIZAREA INFORMATICĂ

Lucrare de licență

**Aplicație Java cu baze de date SQL –
Bibliotecă online**

Coordonator științific

Conf.univ.dr. Radu BORIGA

Absolvent

Antone Andreas

București

2018

CUPRINS

INTRODUCERE

Motivație.....	3
Obiective.....	3
Structura lucrării.....	4
Aplicație practică.....	4

CAPITOLUL I. BIBLIOTECILE ELECTRONICE

1.1. Ce este o bibliotecă electronică.....	5
1.2. Istoria bibliotecilor digitale.....	7
1.3. Biblioteca tradițională versus biblioteca digitală.....	9
1.4. Avantajele și dezavantajele bibliotecilor online.....	10

CAPITOLUL II. TEHNOLOGII FOLOSITE

2.1. Java.....	13
2.2. JavaFX.....	14
2.3. XML și CSS.....	15
2.4. SQL.....	16

CAPITOLUL III. ARHITECTURA APLICAȚIEI

3.1 Aspecte generale ale aplicației.....	18
3.2 Descrierea explicită a aplicației.....	19

CAPITOLUL IV. IMPLEMENTARE

4.1. Pachete și clase.....	29
4.2. Baza de date.....	36

CONCLUZII.....	40
----------------	----

BIBLIOGRAFIE.....	41
-------------------	----

INTRODUCERE

Motivație

Există numeroase motive pentru care avem nevoie de bibliotecile digitale: ar putea face studiul și cercetarea mai ușoare pentru cei de pe băncile școlii și facultății, ar putea ușura presiunile bugetare asupra bibliotecilor, ar putea rezolva problemele noastre din ce în ce mai urgente de conservare sau ar putea ajuta bibliotecile să extindă colecțiile în noile mijloace media. Dar, probabil, cel mai important avantaj ar fi capacitatea lor de a ajuta societatea, de a face mai accesibilă informația, de a-și crește calitatea și diversitatea. Pot bibliotecile digitale să facă asta? Acest lucru va depinde de modul în care finanțăm, reglementăm și gestionăm bibliotecile digitale, noua infrastructură de comunicații și noile tehnologii care le conduc.

Fundamentele care stau la baza acestei lucrări sunt legate de problemele aduse de tehnologie. Mulți dintre noi observăm că oamenii nu mai sunt atât de interesați de lectură. De ce oare? Dependența de telefon, robotizarea Facebook-ului, agitația din zilele noastre, stresul la locul de muncă, bombardarea cu divertisment de proastă calitate, preocupările cu sarcinile zilnice – toți acești factori conduc la scuza: “nu am timp să citesc”.

Mulți dintre noi preferăm ca în drum spre muncă să dăm scroll pe Facebook, să citim noutățile pe site-urile de știri sau pur și simplu să ne deconectăm de restul lumii ascultând muzică în căști. Copii sunt ademeniți de smartphone-uri prin multitudinea de aplicații interactive (jocuri, filmulețe) și prin conexiunea la internet, unde pot găsi orice.

Obiective

De ce în loc să facă toate aceste activități monotone, oamenii de orice vârstă, nu citesc? Este adevărat că pentru a citi ne trebuie pasiune, deoarece este o activitate mai puțin dinamică și mai mult imaginativă, care ne ajută să ne dezvoltăm gândirea, să ne teleportăm într-o altă lume și cel mai important, să ne recreăm.

De ce seara când ajungem acasă deschidem televizorul sau butonăm telefonul, în loc să citim? Tocmai pentru a încerca să evit răspunsuri de genul “nu am timp să ajung la librărie, nu vreau să dau banii pe cărți, nu știu dacă acea carte mă va capta” am realizat o aplicație pentru computere ce prezintă o bibliotecă online. Astfel am vrut să creez un univers al literaturii actualizat la nevoile din zilele noastre.

Structura lucrării

Conținutul efectiv al acestei lucrări este structurat în trei principale capitole ulterioare acestei introduceri și o concluzie a prezentei lucrări.

Capitolul I își propune să prezinte cititorilor acestei lucrări o imagine de ansamblu despre o bibliotecă digitală, prin noțiuni teoretice și un scurt istoric al acestora.

Capitolul II introduce noțiunile teoretice folosite în implementarea aplicației. În secțiunile acestui capitol se vor regăsi detalii despre tehnologiile principale dar și auxiliare folosite la implementarea aplicației.

Capitolul III este destinat prezentării detaliate a interfeței acestei aplicații și a modului ei de utilizare, precum și a pașilor urmați în realizarea bibliotecii online.

Ultimul capitol încheie această lucrare cu concluzii privind utilitatea și performanța aplicației prototip dezvoltată în scop academic și demonstrativ.

Aplicație practică

Prin această aplicație am vrut să creez o bibliotecă online, disponibilă pe laptop sau PC, la care persoane de toate vârstele au acces gratuit. În această comunitate online utilizatorii pot adăuga cărți și materiale în format PDF, pot citi orice carte distribuită de alt utilizator, pot adăuga cărți în lista de favorite.

Deși aplicația implementează toată seria de algoritmi, ea nu își propune să fie o aplicație optimizată pentru a fi comercializată, ci este doar un prototip demonstrativ pentru conceptele teoretice enunțate în lucrare.

CAPITOLUL I. BIBLIOTECILE ELECTRONICE

1.1. Ce este o bibliotecă electronică

În zilele noastre, există o neînțelegere a termenului de bibliotecă digitală - mulți oameni consideră World Wide Web o bibliotecă digitală din cauza înglobării a mii de persoane și mii de documente în mai multe formate și în mai multe versiuni. Totuși, un document se poate găsi online astăzi, și mâine poate fi sters.

Deci, ce este o bibliotecă digitală?

Există multe perspective diferite. De exemplu, pentru majoritatea oamenilor, este o bază de date vastă care oferă acces direct la orice tip de informație pentru persoanele de oriunde. Pentru ingineri și programatori, este o colecție diferită de algoritmi informatici, iar pentru tehnicieni, este încă o aplicație web, în timp ce pentru bibliotecari este un alt pas în domeniul științelor bibliotecii și informațiilor (LIS) și include o colecție de resurse și sisteme create pentru a realiza cu scopuri precise cererile către anumite comunități și grupuri de utilizatori.

Localizarea informațiilor pe internet este inefficientă în raport cu resursele bibliotecilor și nu oferă servicii de înaltă calitate. Mai exact, Clifford Lynch (1997) afirmă: "Internetul nu a fost conceput pentru a sprijini publicarea și recuperarea organizată a informațiilor ca și bibliotecile. A evoluat în ceea ce ar putea fi considerat ca fiind un depozit haotic pentru producția colectivă a "tipografiilor digitale" din lume". Astfel, potrivit lui Clifford Lynch, "Internetul nu este o bibliotecă digitală".

Bibliotecile digitale sunt organizații care furnizează resursele (inclusiv personal specializat), selectează, structurează și oferă acces intelectual la interpretarea, distribuirea, păstrarea integrității și asigurarea persistenței în timp a colecțiilor de date digitale (Schwartz, 2000). Astfel, oamenii pot folosi resursele cu ușurință și în mod econom.

Termenul de bibliotecă digitală a fost folosit pentru prima oară de oamenii de știință, referindu-se la versiunea electronică a unei biblioteci; și adevărul este că termenul bibliotecă digitală nu explică nimic mai mult decât natura colecției sale.

"O bibliotecă digitală este o colecție de reprezentări digitale ale numeroaselor tipuri de medii, cum ar fi documente, imagini și sunetele care sunt stocate într-un depozit de informații și sunt disponibile printr-o rețea locală de calculatoare sau oriunde pe internet." (Slovney, 2004)

Lynch (2005) remarcă faptul că biblioteca digitală poate fi folosită de diverse persoane, cum ar fi ingineri, tehnicieni, oameni de științe sociale și o altă diversitate de oameni de știință, în timp ce Hughes (2004) recomandă că "toată lumea poate căuta, naviga și compara materialele prin modalități utile și creative". În acest fel, crearea unei biblioteci digitale este o soluție foarte bună, dar bibliotecarul E. Rama Reddy subliniază faptul că "Introducerea unei biblioteci digitale doar pentru că există o nouă tehnologie este un lucru greșit".

Biblioteca digitală mai este cunoscută și sub termenii de bibliotecă electronică sau bibliotecă online.

Când ne referim la o bibliotecă digitală o serie de termeni precum "baze de date, fișiere și aplicații multimedia, resurse informaționale online, tehnologii noi de informare și comunicare, rețele de calculatoare, resurse pentru căutarea și găsirea informației, bibliotecă electronică, Internet, Intranet, World Wide Web (www), WAIS (Wide Area Information Services) etc." se subînțeleg. Observăm că toți acești termeni au legătură cu procesele de a construi, prelucra, comunica și utiliza informația. Într-o accepțiune simplă, biblioteca digitală ar trebui să fie, "meta-termenul" care îi reunește.

Intuitiv, înțelegem prin bibliotecă digitală conceptul organizatoric care reunește resurse informaționale în formă electronică, structurate în baze de date, produse și aplicații multimedia, accesibile prin intermediul unor tehnologii de informare și comunicare¹.

"Association of Research Libraries" a identificat următoarele elemente în definițiile bibliotecii digitale:

- nu ne putem raporta la biblioteca digitală ca la o entitate singulară
- tehnologia de care are nevoie o bibliotecă digitală trebuie să fie compatibilă cu resursele celorlalte biblioteci și să asigure accesul la resurse informaționale externe
- legăturile stabilite între mai multe biblioteci electronice și serviciile de informare trebuie să fie transparente pentru utilizatorii finali
- scopul bibliotecilor digitale este asigurarea accesului universal la acestea și la serviciile de informare;
- colecțiile bibliotecilor virtuale nu se limitează la documente text, hypertext sau hypermedia, ele se extind la documente digitale ce nu pot fi reprezentate/distribuite în format tipărit.

¹ Conf. univ. dr. Elena Tîrziman Biblioteca digitală – O încercare de definire

Trei piloni principali sprijină biblioteca digitală: biblioteca electronică, tehnologiile de informare-comunicare și utilizatorul.

Resursele informaționale ale unei biblioteci electronice integrată într-o bibliotecă digitală sunt diverse: ansamblul resurselor digitalizate ale unei biblioteci (cataloge informatizate care reflectă propriile colective, cataloge informatizate ale altor biblioteci, colecții full-text digitalizate) precum și link-uri către alte resurse electronice ce se află într-o oarecare conexiune logică sau organizațională (resurse și instrumente de documentare la distanță).

Tehnologiile de informare și comunicare demonstrează că biblioteca digitală deține ca parte integrantă infrastructura tehnologică ce condiționează procesele de construcție, prelucrare, comunicare și utilizare a informației.

Bibliotecile digitale nu sunt percepute ca fiind entități singulare, deoarece este necesară conectarea obligatorie la rețele și disponibilizarea resurselor informative prin intermediul lor. În acest context informational, cel care "evaluează" biblioteca digitală este utilizatorul.

Transparența serviciilor de informare, interfețele familiare, instrumentele și programele de informare și comunicare folosite de utilizatori arată măsura în care o bibliotecă digitală și-a atins scopul.

Integrând cele 3 componente (resursele informaționale, tehnologiile de informare și de comunicare și utilizatorul) într-un concept organizațional, putem accentua concluzia lui Paul Duguid: *"biblioteca digitală reprezintă contextul care include într-o strânsă dependență colecțiile electronice și instrumentele de management ale informației."*²

1.2. Istoria bibliotecilor digitale

Evoluțiile în tehnologie au adus schimbări critice și un mediu dificil în bibliotecile academice. Introducerea tehnologiei în biblioteci a schimbat modul în care funcționează bibliotecarii, sarcinile și rolurile acestora. În trecut, bibliotecile tradiționale erau clădiri fizice pe care utilizatorii le puteau vizita și unde puteau găsi cărți tipărite. În mod tradițional, bibliotecarii au fost responsabili pentru dezvoltarea colecției, pentru prelucrarea materialelor și furnizarea de servicii fără utilizarea calculatoarelor. În zilele noastre, bibliotecile academice fac uz de tehnologie. Ele oferă nu numai materiale printate, dar și materiale digitale. Introducerea în era digitală a bibliotecilor, a creat nevoia de a computeriza majoritatea operațiunilor. Tehnologia a schimbat modul în care bibliotecarii tradiționali obișnuiau să muncească. În zilele noastre, bibliotecarii folosesc tehnologii pentru achiziționarea,

² Paul Duguid, Report of the Santa Fe Planning Waskshop on Distributed Knowhedge Work Guvirouments.

catalogarea, păstrarea, împrăștierea, furnizarea serviciilor de referință și așa mai departe. Asta nu înseamnă că bibliotecile tradiționale vor dispărea. Bibliotecile tradiționale, ca și clădire vor continua să existe mulți ani de acum încolo și vor sprijini bibliotecile digitale. Prin urmare, bibliotecile tradiționale coexistă cu bibliotecile digitale. Acest lucru înseamnă că bibliotecarii joacă și vor continua să joace un rol semnificativ în biblioteci. Este important pentru bibliotecari să păstreze un echilibru între rolul lor tradițional și rolul digital.

Primii pași către o bibliotecă virtuală au fost făcuți de către utilizatorii calculatoarelor, ei stocându-și electronic anumite documente, înainte de apariția și folosirea internetului. Prin utilizarea calculatoarelor, bibliotecile și-au format în paralel o bibliotecă virtuală, unde au transferat în format electronic edițiile rare sau deteriorate, pentru a proteja distrugerea lor. O bibliotecă virtuală nu se limitează numai la documente ce pot fi tipărite, ci conține și documente audio-vizuale sau documente multimedia organizate în colecții.

Pornind de la biblioteca tradițională și până la biblioteca digitală, întâlnim biblioteca hibrid, și biblioteca electronică. Toate aceste noțiuni, până la un anumit punct au un conținut semantic asemănător. Ele au în vedere procesele de organizare a unui conținut informațional și de utilizare al acestuia, se întrepătrund și se condiționează reciproc.

În cazul bibliotecii tradiționale, accentul cade pe forma tipărită. Biblioteca tradițională îi spune utilizatorului unde se găsește informația pe care o caută, în ce document, fără a-i da direct informația. Biblioteca hibrid este biblioteca în care calculatoarele își fac simțită prezența, în care coexistă documentele tradiționale tipărite, documente audio-vizuale, documente multimedia, iar utilizatorul are posibilitatea de a accesa resursele informaționale prin rețele și în principal prin Internet. Instrumentele de regăsire a resurselor documentare ce sunt puse la dispoziția utilizatorilor se găsesc în format electronic. În componența bibliotecii electronice regăsim documente text, audio-vizuale, multimedia ce sunt transferate pe suport electronic, organizate în colecții și impun condiții specifice de consultare. Biblioteca electronică permite utilizatorilor accesul online la catalogul informatizat, dar și la alte documente secundare de informare (ex: bibliografii) și poate îngloba sau partaja cu alte biblioteci electronice resurse informaționale electronice ce există sau nu în bibliotecă. Biblioteca electronica constituie germenul bibliotecii digitale, ea permițând accesul la resurse electronice care nu se găsesc în colecțiile proprii.

1.3. Biblioteca tradițională versus biblioteca digitală

Potrivit lui Hemlata & Meena (2013) diferențele de bază dintre bibliotecile digitale și bibliotecile tradiționale sunt următoarele:

- ✓ biblioteca digitală este un obiect digital și poate fi îmbinată cu altele.
- ✓ biblioteca digitală oferă acces mai rapid la informații, deoarece furnizează un motor avansat de căutare și hyperlink-uri pentru navigare.
- ✓ biblioteca digitală permite gestionarea facilă a unor cantități mari de date, deoarece acestea oferă acces la o cantitate amplă de resurse de informații.
- ✓ colecțiile bibliotecilor digitale sunt actualizate permanent, pentru a oferi informații utile și ușor de înțeles pentru utilizatori.
- ✓ biblioteca digitală sprijină procedurile de învățare formală și informală, prevede accesul la sursele de învățământ primar și secundar.
- ✓ biblioteca digitală poate fi accesată de orice utilizator din orice loc, deoarece se bazează pe un sistem ușor de utilizat

Aceste caracteristici sunt definite următoarele etape:

(1) Selecția & Achiziția: selectarea materialului adecvat pentru a fi digitizat (adică colecții unice, vechi) sau / și achiziționarea de lucrări digitale originale (adică cărți electronice, jurnale, seturi de date).

(2) Organizarea: atribuirea metadatelor (informații bibliografice) fiecărui material, fiind adăugate în colecția de biblioteci digitale.

(3) Indexarea și stocarea: indexarea și stocarea documentelor digitale și a metadatelor pentru căutarea eficientă și recuperarea datelor.

(4) Căutarea & Găsirea: dezvoltarea unei conexiuni web pentru căutarea în biblioteca digitală prin intermediul metadatelor adecvate și a protocoalelor de schimb de informații. Acesta este prezentat, de obicei, ca pagină de limbă Hypertext Markup Language (HTML) pentru utilizatorii bibliotecii.

O idee generală este că o bibliotecă digitală este mai ieftină decât cea tradițională, dar crearea unei biblioteci digitale înseamnă crearea unei organizări complete a tehnologiilor și a oamenilor și este un proiect costisitor. Instrumentele de digitizare nu sunt ieftine. Finanțarea bugetară joacă un rol esențial în dezvoltarea bibliotecilor digitale, pentru că nu contează ceea ce profesioniștii bibliotecari vor să facă, sau care sunt nevoile utilizatorului, banii sunt sursa tuturor deciziilor (Kastellec, 2012).

1.4. Avantajele și dezavantajele bibliotecilor online

O bibliotecă virtuală nu este condiționată de noțiunea spațio-temporală. Ea stochează în "rafturile" sale electronice un număr imens de volume, documente, materiale, care nu ar încăpea într-o bibliotecă tradițională. Avantajul oferit de acest gen modern de bibliotecă este faptul că poți găsi cu foarte mare ușurință, nu numai cartea căutată, dar și pasajele dorite din conținutul ei. Lectura se poate face indiferent de oră, după cum este dispus cititorul, fără program de acces. Cartea palpabilă trebuie ținută în mâini în timpul citirii și apoi este lăsată pe masă sau pe noptieră, în timp ce cartea virtuală este invizibilă, nu are copertile roase sau decolorate, nici pagini îngălbenite sau rupte.

1.4.1. Avantajele bibliotecilor online

Dacă ne-am întoarce în trecut, și am fi prin anul 2000, am putea concluziona următoarele:

Până acum, digitizarea a fost descrisă ca fiind ceva care evită problemele. În plus, electronica poate oferi multe avantaje bibliotecilor. Acestea includ atât acces mai bun la materialele tradiționale, conservarea mai ușoară, cât și extinderea colecțiilor bibliotecilor.

După cum mulți au descoperit, capacitatea de căutare oferă un avantaj enorm pentru materialele electronice atunci când este disponibilă o versiune Ascii. Căutările online au înlocuit, timp de câțiva ani cărțile tipărite, iar acum, în publicații, cum ar fi Baza de date a poeziei englezești Chadwyck-Healey, putem găsi materialele electronice disponibile cititorilor.

Electronica oferă și alte câteva avantaje: o singură copie electronică poate fi accesată dintr-un număr mare de locații și de mulți utilizatori simultani (presupunând că dreptul de autor este acordat). Copiile pot fi livrate cu viteză electronică și este posibilă ajustarea materialelor la comoditatea cititorului (de exemplu, mărimea caracterelor pentru cei cu vedere limitată sau micșorarea lor pentru a încadra foaia pe ecran mai mic). Deoarece cititorii obțin un afișaj al obiectului, în loc să îndepărteze obiectul fizic, rata pierderilor prin furt poate fi mult redusă. Din păcate, deși biblioteca poate fi imună la pierderea propriilor copii prin furt, în lumea digitală există un tip diferit de furt în care titularului cărții îi este furat dreptul de autor.

Un alt avantaj important al bibliotecilor digitale este conservarea. Informațiile digitale pot fi copiate fără eroare. Drept urmare, conservarea într-o lume digitală nu depinde de a avea un obiect permanent și de a-l păstra sub pază, ci de capacitatea de a face mai multe copii, presupunând că cel puțin una va supraviețui. Riscul major al obiectelor digitale nu este, de obicei, deteriorarea fizică, ci uzura tehnologică a dispozitivelor pentru a le citi. Banda magnetică cu scanare elicoidală este probabil singurul format care se poate uza înainte ca acesta să devină caduc. Durata de viață a cartușelor magneto-optice, a benzii magnetice liniare și, cu

siguranță, a CD-ROM-ului este de așteptat să fie de decenii, în timp ce durata de viață a dispozitivelor de citire este de numai aproximativ o decadă (cu excepția CD-ROM-ului). Dar, presupunând un program rezonabil de reconfortare și o încercare rezonabilă de a evita achiziționarea de dispozitive care au un scop special, bibliotecile se pot aștepta ca informațiile digitale să fie păstrate cu o deteriorare mai redusă decât hârtia.

Stocarea digitală permite, de asemenea, bibliotecilor să extindă gama de materiale pe care le pot oferi utilizatorilor lor. De exemplu, și casetele audio și înregistrările de vinil aduc probleme bibliotecilor; nu rezistă unui număr mare de redări fără a se deteriora. Digitizarea sunetului poate produce un format (CD-ul audio) care este mult mai sigur de folosit. Materialul digital poate de asemenea să permită accesul la fotografii fragile, într-o zi la benzi video și la noile tipuri de materiale multimedia create doar pe calculator și care pur și simplu nu vor avea un echivalent în niciun format tradițional.

Dacă ne raportăm la anul 2018, observăm cât de mult a evoluat biblioteca online.

Cărțile electronice (în engleză E-book) au fost concepute pentru a fi citite pe calculator (inițial), dar odată cu evoluția tehnologiei s-au dezvoltat și pe alte dispozitive cum ar fi smartphone-ul, book rider, laptop sau tabletă. Ele pot fi văzute ca o alternativă a cărților obișnuite, ele putând fi ușor găsite în mediul online și citite imediat.

Primele cărți electronice apărute erau destinate unui anumit tip de public și tratau un număr restrâns de subiecte, dar odată cu expansiunea internetului, ebook-urile au devenit o sursă importantă de informații și răspunsuri la o gamă mare de probleme și întrebări, tot mai multe persoane având acces la aceste cărți digitale.

O carte electronică poate fi citită de pe orice calculator, fiind disponibilă în format PDF, citirea ei neimplicând programe adiționale. Atât în lumea reală cât și în lumea virtuală există așa numitele “edituri” care se ocupă cu publicarea cărților electronice. Părerile scriitorilor sunt împărțite: unii nu au nimic împotrivă în crearea unei versiuni electronice a cărților publicate în format fizic, în timp ce alții consideră că disponibilitatea electronică nu ar oferi prea multe avantaje. Cărțile electronice din zilele noastre sunt disponibile în foarte multe domenii, aproape în toate, și au un public foarte numeros. Acesta este motivul pentru care vânzările cresc într-un ritm foarte alert. Unele cărți sunt gratuite, se pot găsi foarte ușor pe Internet, se poate descărca pe calculator/telefon, iar altele sunt contra-cost, accesarea lor necesitând o sumă de bani.

E-bookurile, cărțile care au câștigat popularitate în ultimii ani, prezintă o serie de avantaje cum ar fi: economisirea timpului, accesibilitate, comoditate. Când se cumpără o carte electronică, nu mai este necesară deplasarea la magazin pentru achiziția ei, sau nu mai este

nevoie să fie așteptată livrarea prin curierat. O carte electronică poate fi cumpărată foarte simplu și poate fi citită imediat, la un preț mai mic decât al unor cărți obișuite, sau chiar pot fi procurate gratuit.

1.4.2. Dezavantajele bibliotecilor online

O monedă are două fețe, așa și cărțile electronice au un revers al banului, reprezentat prin dezavantaje. Unul dintre ele ar fi necesitatea unui calculator, al unui telefon mobil, al unei tablete sau oricărui alt dispozitiv capabil să redea conținutul cărții într-un format ce poate fi citit cu ușurință. În plus, riscul pierderii cărții dacă dispozitivul se strică, este virusat sau dacă unele fișiere au fost șterse accidental din sistem, este mare. Iar pirateria este al treilea dezavantaj. O carte electronică poate fi pirată (la fel ca orice alt program sau aplicație), adică copiată sau modificată fără acordul autorului, astfel imaginea acestuia poate avea de suferit și implicit veniturile lui.

Existența bibliotecii virtuale a simplificat procesul de achiziționare, depozitare și distribuire a cărților, reducând semnificativ numărul bibliotecarilor angajați.

CAPITOLUL II. TEHNOLOGII FOLOSITE

2.1. Java

Tehnologia principală pentru codarea aplicației este Java, motivul alegerii acestei tehnologii fiind faptul că Java este un limbaj de programare orientat-obiect de nivel înalt și rulează pe mai multe sisteme de operare cum ar fi: Windows, Mac Os și mai multe versiuni de UNIX.

Limbajul de programare Java a fost inițial conceput de James Gosling, dezvoltat în cadrul companiei Sun Microsystems (acum filială Oracle), și lansat în 1995 ca o componentă centrală a platformei Sun Microsystems' Java (Java 1.0 [J2SE]).

Cea mai recentă versiune a limbajului Java este Java SE 8 (Java Standard Edition 8). Au fost create mai multe configurații pentru a se potrivi diferitelor tipuri de platforme (de ex: J2EE pentru aplicații de tipul Enterprise, J2ME pentru aplicații mobile etc.) datorită dezvoltării limbajului Java și popularității sale pe scară largă.

Limbajul Java are următoarele caracteristici:

- **orientat-obiect** – În Java, totul este un obiect. Java poate fi extins cu ușurință, deoarece se bazează pe modelul obiect, conținând toate aspectele legate de programarea orientată pe obiecte (clase, obiecte, încapsulare, moștenire, transmitere de parametri, modificatori de acces și biblioteci).
- **independent de platformă** – Java este independent de platformă, deoarece același cod poate fi executat de orice sistem care deține o JVM (mașină virtuală Java). Mașina virtuală Java se instalează odată cu limbajul Java, iar spre deosebire de multe alte limbaje de programare, codul Java este interpretat de mașina virtuală.
- **simplu, sigur și robust** – Java a fost conceput pentru a fi ușor de învățat, simplitatea fiind o caracteristică primară. Pentru a programa în Java nu ai nevoie de mult antrenament, iar dacă înțelegi conceptele de bază ale programării orientate pe obiect va fi ușor să îl stăpânești. Siguranța și robustețea sunt alte calități importante ale limbajului, acesta permitându-ne dezvoltarea de aplicații fără viruși și fără erori, Java depunând eforturi considerabile pentru eliminarea situațiilor predispuse la erori.
- **multi-threaded (concurrent)** – În Java putem scrie programe care rulează diferite task-uri simultan. Această caracteristică a limbajului permite dezvoltarea de aplicații interactive care rulează fără probleme.

- **dinamic** – Java este considerat mai dinamic decât cel mai puternic limbaj orientat pe obiecte (C++), deoarece este conceput să se adapteze la un mediu în continuă dezvoltare.

Java nu folosește pointeri și alocă memorie doar la execuție, fapt pentru care programele scrise în Java nu pot accesa memoria de cod, memoria stivă/heap sau alte secțiuni protejate de memorie.

2.2. JavaFX

JavaFX este o bibliotecă Java utilizată pentru a construi aplicații cu interfețe grafice avansate. Aplicațiile scrise cu această librărie pot rula în mod consistent pe mai multe platforme. Aplicațiile dezvoltate utilizând librăria JavaFX pot rula pe diferite tipuri de dispozitive cum ar fi calculatoare personale, telefoane mobile, tablete, televizoare, etc.

În general, pentru a dezvolta aplicații cu interfață grafică în limbajul Java, programatorii se folosesc de biblioteci precum “Advanced Windowing Toolkit” sau “Swing”. Odată cu apariția librăriei JavaFX, programatorii pot dezvolta în prezent aplicații de tip GUI (eng. Graphical User Interface) cu un conținut bogat datorită facilităților cu care vine această librărie, întrucât atât componenta server (nivelul de logică a aplicației și de acces la date) cât și componenta client (nivelul de prezentare) vor fi dezvoltate utilizând același limbaj de programare.

JavaFX este un limbaj de programare ușor de utilizat deoarece este derivat din limbajul Java, acesta aducând noi optimizări complexității sistemelor și costurilor datorită componentei client-server dezvoltate de același limbaj.

Limbajul JavaFX are următoarele caracteristici:

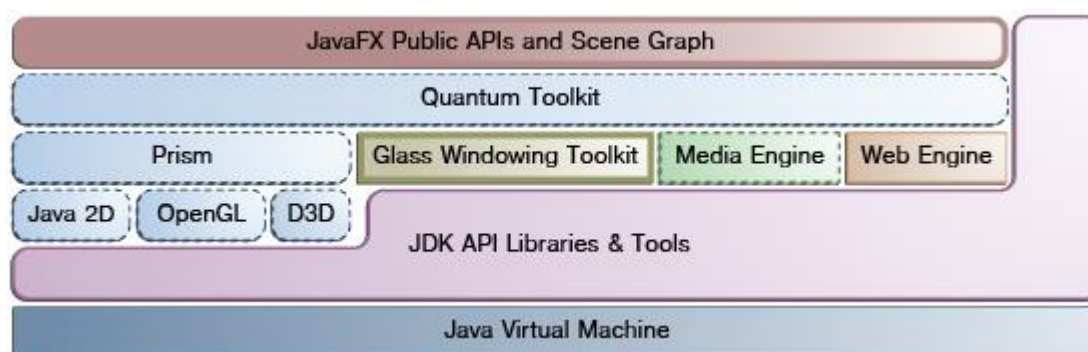
- **scris în Java** – JavaFX este scris în Java și este disponibil pentru limbajele ce se pot executa pe mașina virtuală Java (Java, Groovy și JRuby).
- **Scene Builder** – JavaFX oferă o aplicație numită Scene Builder cu ajutorul căreia putem modela ușor și rapid interfețele aplicației.
- **FXML** – JavaFX ne oferă un nou limbaj cunoscut ca FXML, acesta fiind un limbaj de anotare bazat pe limbajul XML. Scopul acestui limbaj este de a dezvolta interfețe grafice prietenoase cu utilizatorul.
- **librărie grafică** – JavaFX ne pune la dispoziție o librărie grafică integrată pentru grafica 2D și 3D.

- **temă predefinită** – JavaFX folosește ca temă predefinită pentru toate aplicațiile dezvoltate tema numită Moderna, schimbarea temei făcându-se foarte ușor prin apelarea metodei `setUserAgentStylesheet(String url)` în metoda `start()`.

Arhitectura limbajului JavaFX

Platforma JavaFX deține o arhitectură ce se bazează pe o stivă de componente, transparentă pentru programator. Aceasta constituie motorul Quantum Toolkit care rulează codul propriu-zis și înglobează motorul grafic performant(Prism), sistemul eficient de ferestre de dimensiuni mici (Glass), motorul pentru redarea conținutului multimedia și motorul pentru integrarea conținutului internet.

Imaginea 2.2. Arhitectura limbajului
JavaFX



Sursa: <http://aipi2015.andreirosucojocaru.ro>

2.3. XML și CSS

XML este prescurtarea de la Extensible Markup Language și este un limbaj de tip markup bazat pe text, derivat din Standard Language Generalized Markup Language (SGML). Etichetele XML identifică datele și se utilizează pentru a stoca și a organiza datele, în loc să precizeze modul de afișare a acestora precum tagurile HTML, care sunt folosite pentru a afișa datele. XML nu a fost creat pentru a înlocui HTML, acesta doar introduce noi caracteristici.

Cele mai importante caracteristici ale XML-ului sunt:

- **XML este extensibil** – XML ne permite să creăm noi etichete sau limbi auto-descriptive, care să se potrivească cu aplicația noastră
- **XML transportă datele, nu le prezintă** - XML ne oferă posibilitatea să stocăm datele indiferent de modul în care acestea vor fi prezentate

- **XML este un standard public** - XML a fost dezvoltat de o organizație numită Consorțiu World Wide Web (W3C) și este disponibil ca standard deschis.

CSS este prescurtarea de la “Cascading Style Sheet” și este folosit pentru a controla stilul unei pagini web într-un mod ușor și simplu, iar cu ajutorul librăriei JavaFX, putem folosi CSS pentru a controla stilul aplicațiilor Java.

Folosind CSS, putem controla culoarea textului, stilul fonturilor, distanța dintre paragrafe, modele de aspect, culoarea fundalului și o varietate largă de efecte.

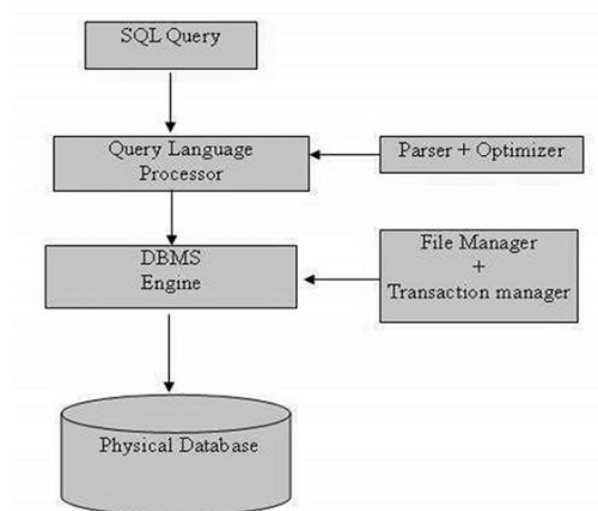
Avantajele CSS-ului sunt:

- **Economisire de timp și spațiu:** Putem să scriem o singură dată un stil CSS și apoi să reutilizăm același stil pentru mai multe pagini din aplicație sau putem aplica un stil pentru fiecare element. Datorită reutilizării stilurilor CSS vom avea mai puțin cod în aplicație, iar paginile se vor încărca mai rapid.
- **Întreținere ușoară:** Pentru a face o schimbare globală, schimbăm pur și simplu stilul și toate elementele din aplicație vor fi actualizate automat.
- **Compatibilitatea cu mai multe dispozitive:** Stilurile sunt optimizate și recunoscute de diferite tipuri de dispozitive
- **Folosirea CSS-ului fără conexiune la internet:** CSS poate să stocheze stilurile la nivel local cu ajutorul unei aplicații offline. Folosind acest lucru, putem vizualiza aplicația offline. Cache-ul asigură, de asemenea, o încărcare mai rapidă și o performanță globală mai bună a aplicației.

2.4. SQL

SQL este un limbaj de programare bazat pe algebra relațională, având ca scop manipularea datelor. Cu acest limbaj putem adăuga, modifica și șterge scheme de date, putem interoga, adăuga, șterge și actualiza datele și putem controla accesul la date. SQL a devenit un standard în domeniu, fiind cel mai popular limbaj utilizat pentru crearea, modificarea, regăsirea și manipularea datelor de către SGBD-urile (Sistemele de Gestiune a Bazelor de Date) relaționale. Numele său provine de la Structured Query Language (SQL) care înseamnă limbaj de interogare structurat. SQL este un limbaj standard ANSI (eng. American National Standards Institute), dar există multe versiuni ale acestuia.

Imaginea 2.4. Arhitectura SQL



Sursa: <https://tutorialeinmd.wordpress.com>

Avantajele limbajului SQL sunt:

- Permite utilizatorilor să acceseze date în sistemele de gestionare a bazelor de date relaționale.
- Permite utilizatorilor să descrie datele.
- Permite utilizatorilor să definească datele într-o bază de date și să le manipuleze.
- Permite încorporarea în alte limbaje utilizând module SQL, librării și precompilatoare.
- Permite utilizatorilor să creeze și să șteargă baze de date și tabele.
- Permite utilizatorilor să creeze vizualizări, proceduri stocate și funcții în baza de date.
- Permite utilizatorilor să stabilească permisiuni pe tabele, proceduri și vizualizări.

CAPITOLUL III. ARHITECTURA APLICAȚIEI

3.1 Aspecte generale ale aplicației

În lucrarea de față am abordat tema bibliotecilor digitale. Am ales aceasta temă deoarece tehnologia a evoluat foarte mult în zile noastre, iar o bibliotecă digitală este necesară. În țara noastră bibliotecile digitale sunt rare sau nu sunt bine alcătuite, motiv pentru care acest domeniu este foarte atractiv.

Lucrarea mea s-a axat doar pe realizarea unei biblioteci digitale cât mai ușor de utilizat, fiind utilă pentru unitățile de învățământ din țara noastră, dar și pentru comunitățile de pe internet care sunt dispuse să împartă cu ceilalți diferite cărți și publicații.

Aplicația este implementată în limbajul de programare Java și folosește diferite librării disponibile acestui limbaj pentru a da stabilitate aplicației și un aspect cât mai plăcut. Librăria de bază a acestei aplicații este ”JavaFx”, dezvoltată de cei de la Oracle, aceasta axându-se pe designul aplicațiilor programate în Java, contribuind foarte mult la crearea unei interfețe prietenoase cu utilizatorul și ușor de înțeles.

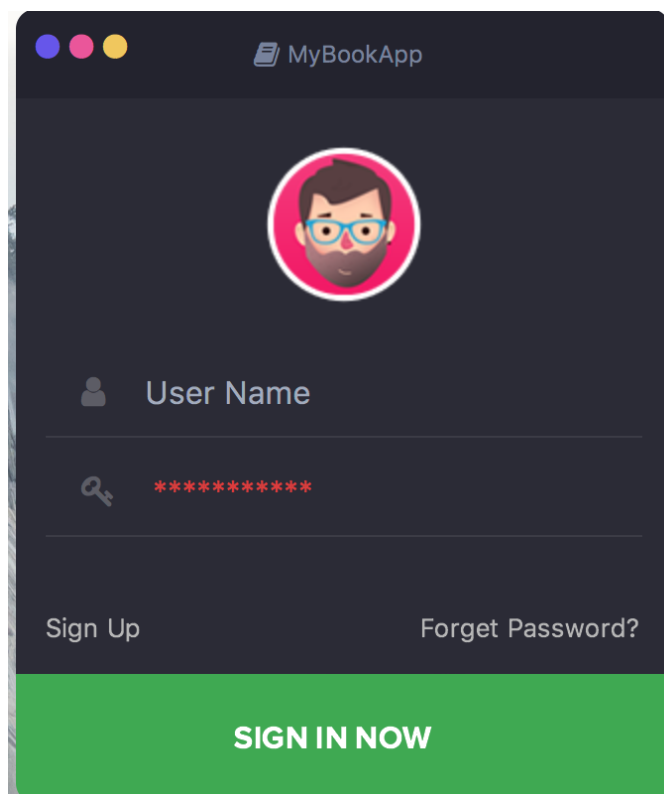
Aplicația lucrează cu o bază de date de tip SQL care stochează informațiile utilizatorilor și cărțile încărcate de aceștia. În baza de date am mai stocat și o serie de 31 de maxime și citate motivaționale, acestea fiind vizibile pe prima pagină a aplicației și actualizându-se în fiecare zi a lunii. De asemenea, am mai stocat în baza de date și problemele aplicației pe care utilizatorii le întâlnesc și le raportează cu ajutorul unei pagini special amenajate pentru acest lucru, precum și sugestiile și îmbunătățirile pe care aceștia le propun.

Interfața grafică a aplicației este complexă, dar intuitivă, aplicația fiind ușor de folosit de către orice persoană care o utilizează, nefiind necesară o instruire despre utilizarea aplicației și funcționalitatea acesteia. Aplicația conține două scenarii mari: scenariul inițial care se prezintă ca o fereastră de dimenisuni mici disponibilă pentru autentificare utilizatorului, recuperarea parolei sau înregistrarea unui cont nou și scenariul principat unde utilizatorul poate să răsfoiască diferite cărți, să contribuie la dezvoltarea aplicației prin adăugarea unor noi cărți, raportarea unor erori sau nereguli din aplicație ș.a.m.d.

3.2 Descrierea explicită a aplicației

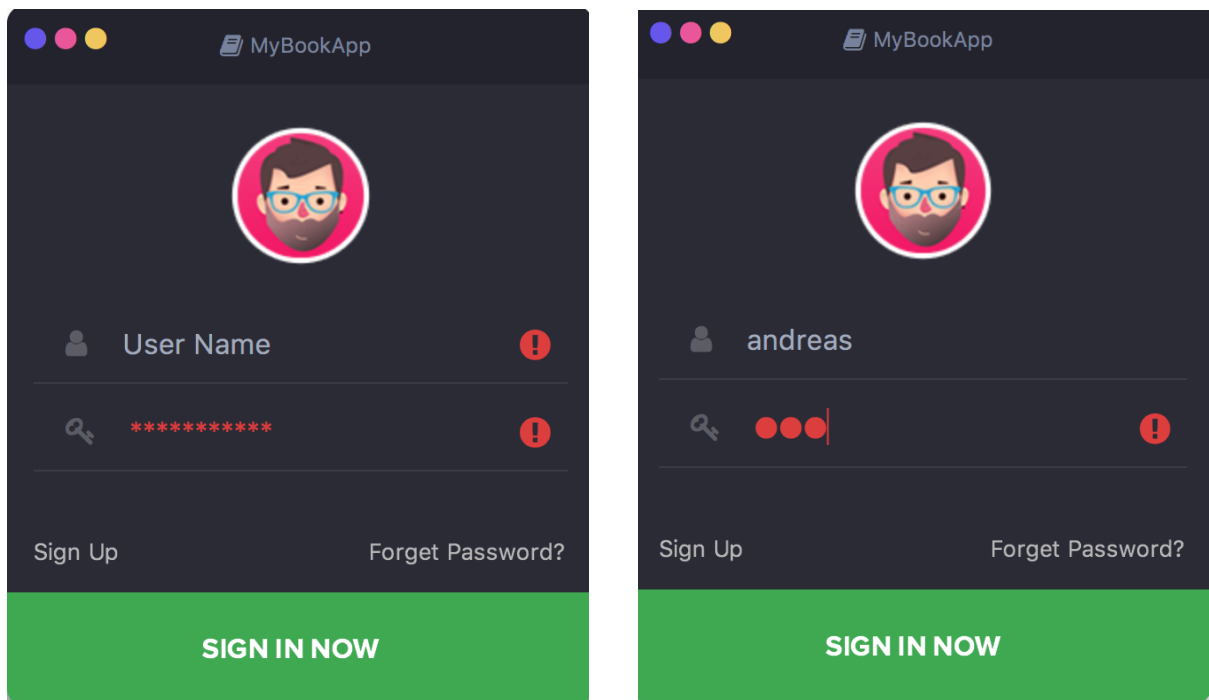
În acest subcapitol vor fi detaliate diferite scenarii ale aplicației însoțite de o scurtă descriere a acestora, punând accent pe utilitatea interfeței grafice.

În momentul rulării programului pe ecranul utilizatorului va apărea primul scenariu al aplicației, care poate fi observat mai jos.



Imaginea 3.2.1 - Pagina de autentificare

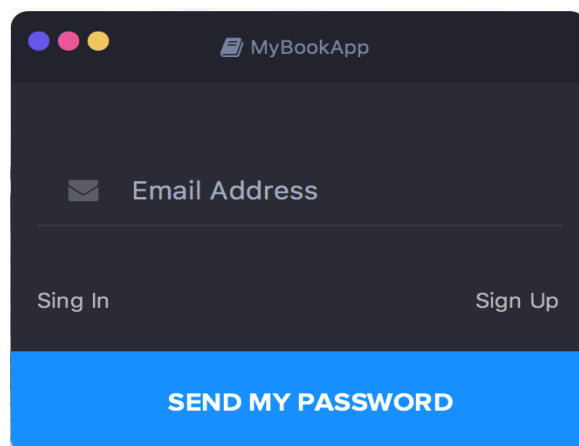
Cu ajutorul scenariului inițial utilizator poate intra în meniul principal al aplicației folosind o autentificare clasică bazată pe un nume de utilizator și parolă. După cum se poate observa în imaginea 3.2.1 aplicația are o interfață intuitivă, iar pentru autentificarea în aplicație se va folosi câmpul pe care este afișat mesajul "User Name" însoțit de pictograma utilizator, iar pentru parolă se va folosi câmpul cu stelute însoțit de pictograma cheie/parolă. După completarea celor două câmpuri, se va acționa butonul de culoare verde numit "SIGN IN NOW" care poate conduce utilizatorul în următoarele trei scenarii: scenariul curent însoțit de pictograme cu semnul exclamării vizibile în dreptul câmpurilor utilizator și parolă, către interfața principală a aplicației sau către un mesaj de eroare care avertizează utilizatorul că datele introduse sunt incorecte sau nu există în baza de date.



Imaginea 3.2.2 - Pagina de autentificare însoțită de alerte pentru câmpurile incomplete

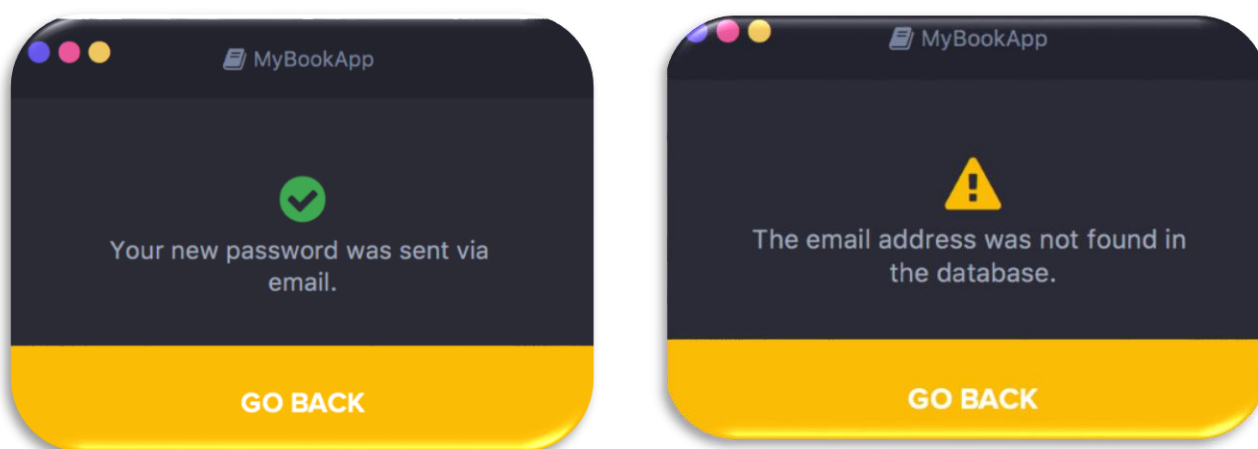
În imaginea 3.2.2 se pot observa pictograme cu semnul exclamării care pot apărea în momentul acționării butonului de logare. În prima captură au apărut două pictograme cu semnul exclamării deoarece informațiile necesare pentru autentificare lipsesc, iar în a doua captură a apărut semnul exclamării doar în dreptul câmpului parolă deoarece informația nu este completă (parola trebuie să fie alcătuită din minimum 6 caractere și s-au introdus doar 3 caractere).

Tot cu ajutorul interfeței inițiale utilizatorul poate reseta parola în cazul în care nu și-o mai amintește sau își poate crea un cont nou.



Imaginea 3.2.3 - Pagina pentru recuperarea parolei

Scenariul din imaginea 3.2.3 este folosit în momentul în care nu ne mai amintim parola și vrem să o recuperăm. Acesta este foarte intuitiv, ajutând astfel utilizatorul să-și recupereze parola doar prin introducerea adresei de email pe care s-a creat contul în câmpul însoțit de pictograma email și pe care este afișat textul "Email Address" și prin acționarea butonului albastru numit "SEND MY PASSWORD". În spatele interfeței stau la bază diferite subprograme Java care verifică dacă formatul adresei de email este corect și dacă adresa se află în baza de date. Algoritmul de recuperare al parolei poate genera la rândul său următoarele trei scenarii: adresa de email nu este într-un format corespunzător (lipsește simbolul "@", domeniul/subdomeniul web sau terminația acestuia), adresa de email nu se află în baza de date și parola se resetează prin trimiterea unui email.



Imaginea 3.2.4 - Mesajele de alertă primite pe pagina de recuperarea a parolei

MyBookApp - Recover your password

To: Andreas Antone

Someone, hopefully you, requested that the password for the account associated with this email address.

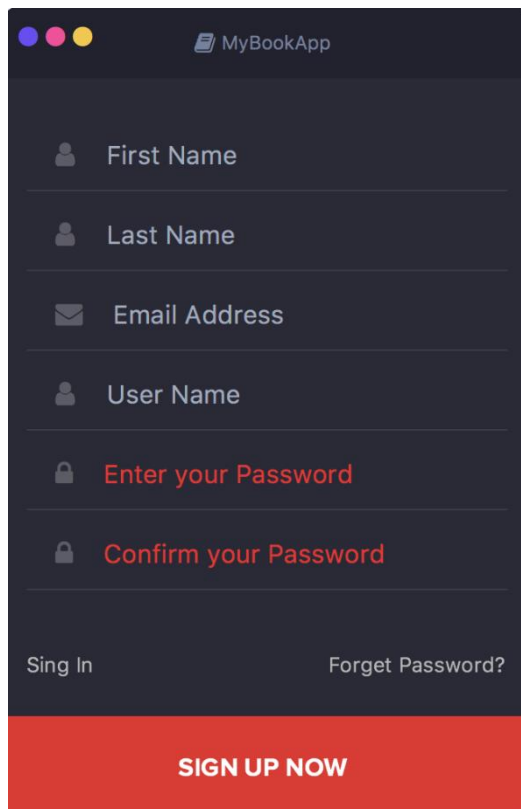
Your new password is: 3fec0456ebdd4dd8

Please do not reply.

Imaginea 3.2.5 - Exemplu de email trimis pentru recuperarea parolei

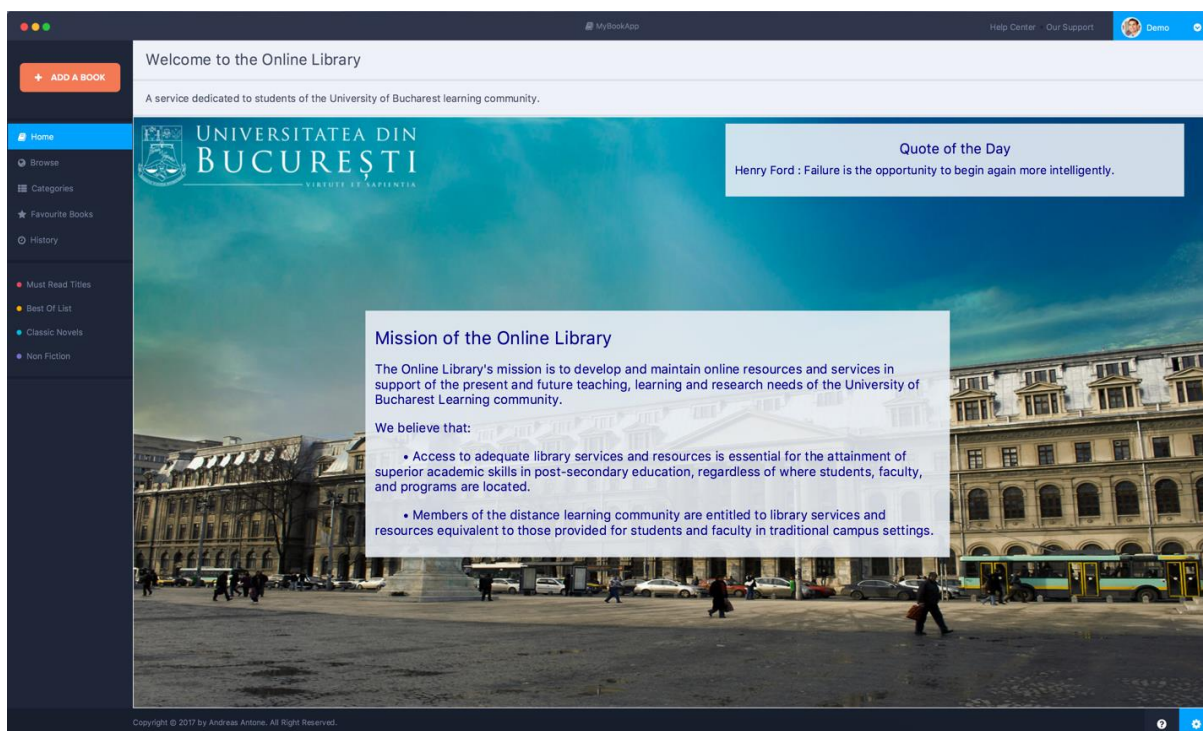
Următorul pas este detalierea ultimului scenariu al interfeței inițiale și anume cel pentru crearea unui cont nou. În opinia mea, acesta este cel mai important scenariu din interfața inițială deoarece pentru a folosi celelalte două scenarii este nevoie de un cont care poate fi creat numai cu ajutorul acestui scenariu.

Crearea unui cont se face completând toate câmpurile sugestive din imaginea 3.2.6. Trebuie menționat că pentru crearea unui cont, suntem obligați să completăm toate câmpurile prezentate în aplicație, iar adresa de email trebuie să fie unică și validă, adică nu se pot crea multiple conturi folosind aceeași adresă de email.

The image shows a mobile application interface for registration. At the top, there are three colored circles (purple, pink, yellow) and the text "MyBookApp". Below this, there are six input fields, each with a small icon to its left: a person icon for "First Name", a person icon for "Last Name", an envelope icon for "Email Address", a person icon for "User Name", a lock icon for "Enter your Password", and a lock icon for "Confirm your Password". The password fields have red text. Below the input fields, there are two links: "Sing In" and "Forget Password?". At the bottom, there is a large red button with the text "SIGN UP NOW" in white capital letters.

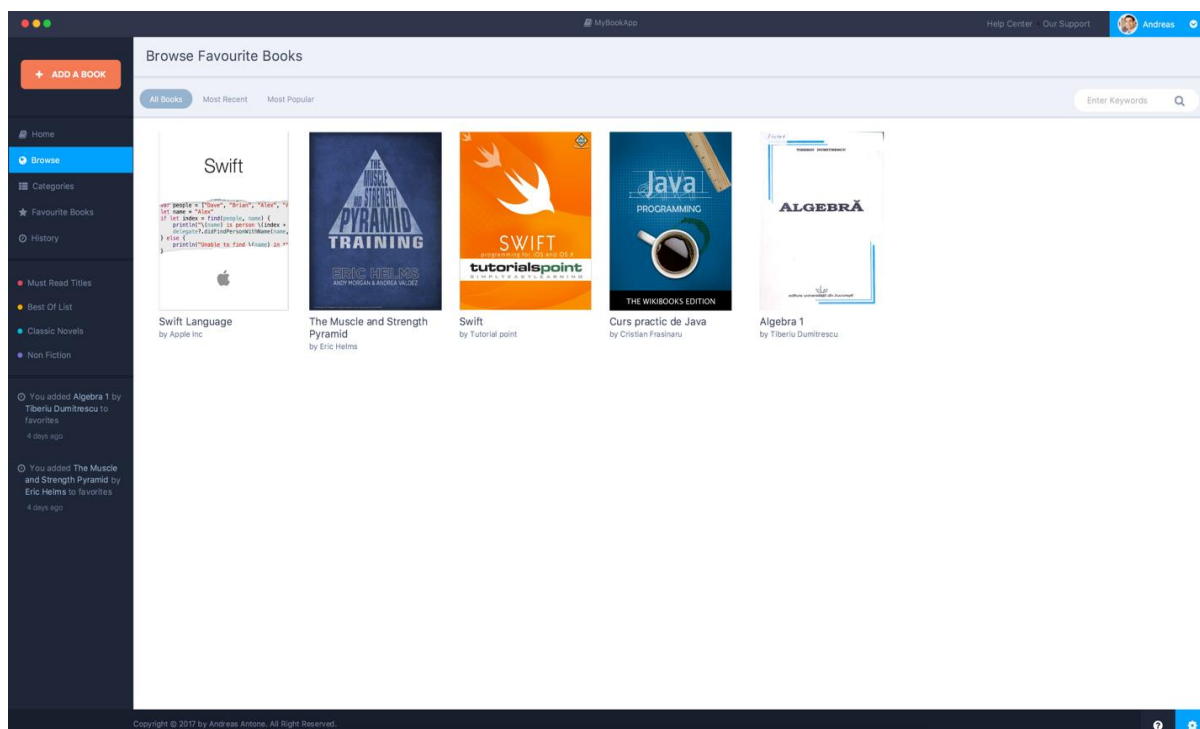
Imaginea 3.2.6 - Pagina pentru crearea unui cont nou

În continuare vom detalia prima pagină din interfața principală. Pagina principală denumită "Home" este pagina cu care avem contact după ce ne-am autentificat în aplicație și reprezintă o pagină de prezentare. Această pagină este alcătuită dintr-o imagine de fundal cu Universitatea din București și diferite mesaje pentru utilizator. În această pagină întâlnim și maxima sau citatul zilei și o scurtă descriere despre misiunea pe care această aplicație și-o propune.



Imaginea 3.2.7 - Pagina "Home"

În imaginea 3.2.8 ne este prezentată pagina denumită "Browse", unde utilizatorul poate să răsfoiască diferite cărți și materiale. Rolul paginii este de a prezenta cărțile din baza de date și de a-i permite utilizatorul răsfoirea, sortarea și căutarea acestora. Din imagine putem observa că în partea din stânga-sus avem trei grupuri pentru sortarea cărților (toate cărțile, cele mai recente, cele mai populare), iar în partea din dreapta-sus avem o căsuță de căutare. Cărțile vizibile în această pagină sunt însoțite de o imagine care reprezintă coperta acestora, precum și de titlul și autorul cărții.



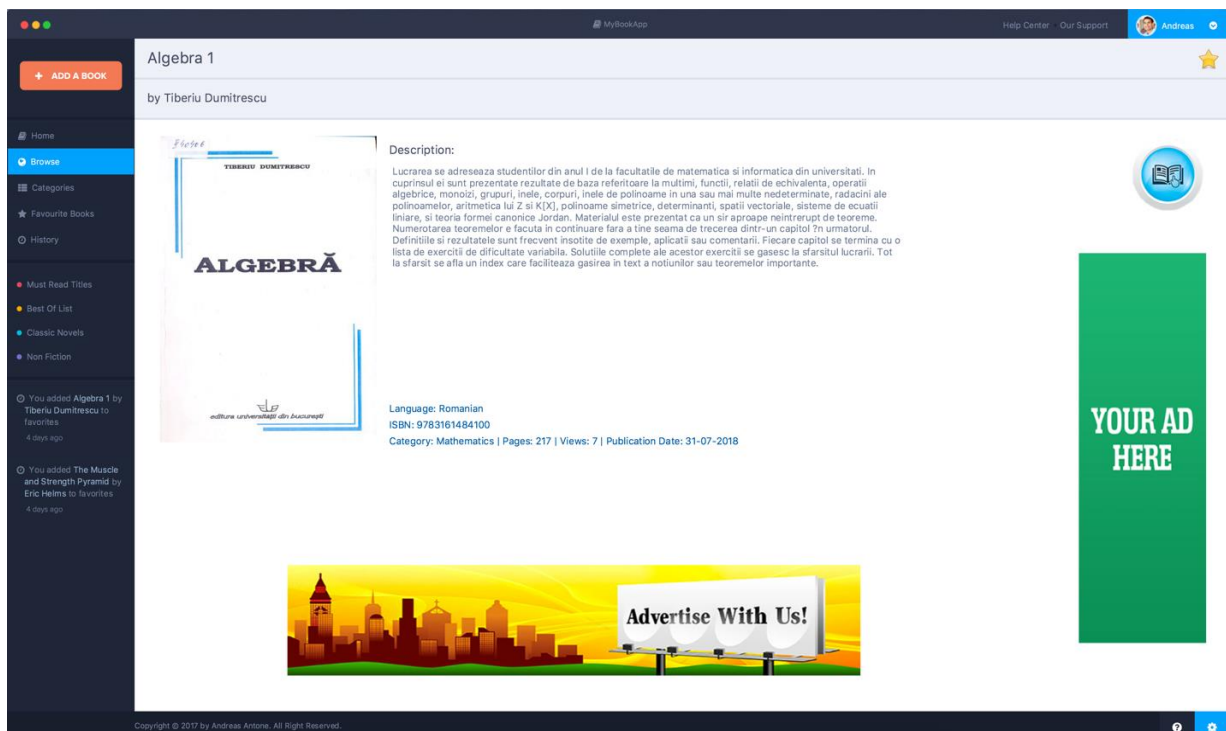
Imaginea 3.2.8 - Pagina "Browse"

În cele ce urmează vom vorbi despre pagina de prezentare a unei cărți. Aceasta este alcătuită de toate elementele care definesc cartea și de anumite funcții ale aplicației.

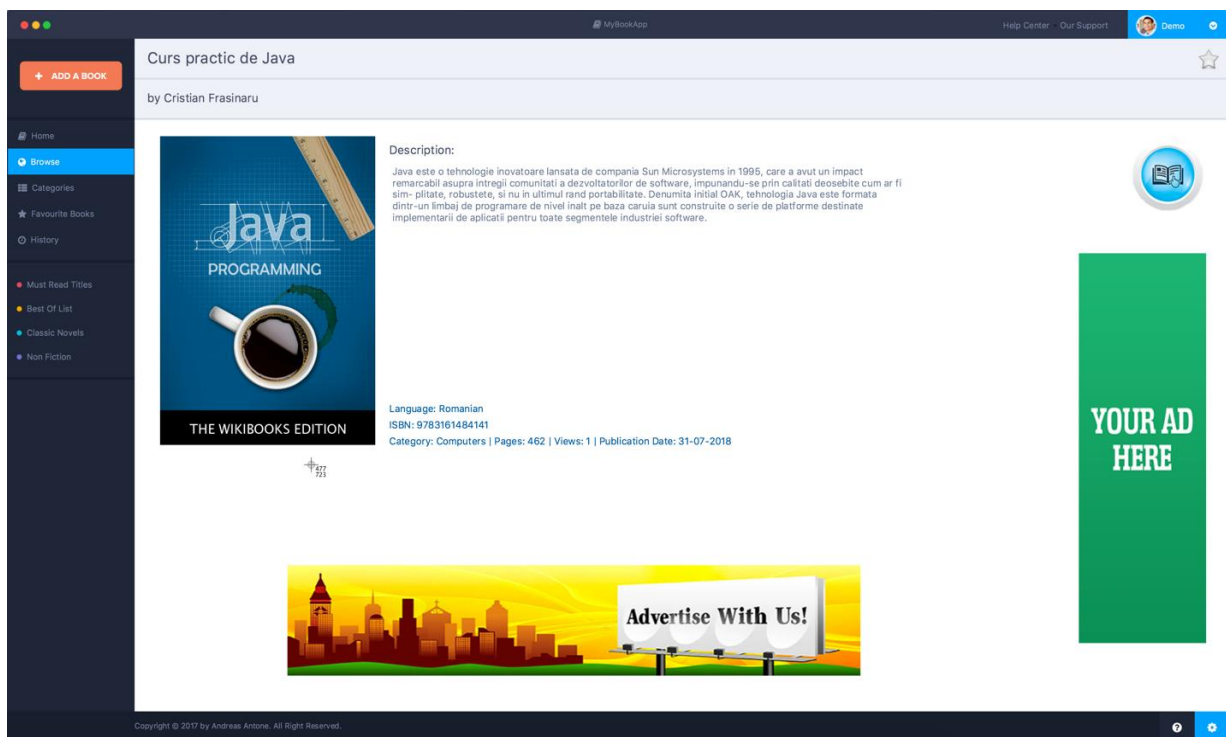
În antetul paginii de prezentare a unei cărți vom avea întotdeauna vizibile în partea din stânga titlul și autorul cărții. Tot în antetul paginii vom găsi butonul "favourite" simbolizat prin pictograma steluță și localizat în partea dreaptă a antetului. Rolul butonului este de a marca o carte ca fiind favorită, făcând-o astfel vizibilă pe pagina cărților favorite despre care vom vorbi mai târziu. Trebuie menționat faptul că acest buton își schimbă culoarea din gri în galben după ce cartea a fost adăugată la favorite. Cartea prezentată în imaginea 3.2.9 este adăugată la favorite deoarece butonul "favorite" este de culoare galbenă, iar cartea din imaginea 3.2.10 nu este adăugată la favorite deoarece butonul are culoare gri.

În interiorul paginii vom avea în partea stângă coperta cărții, iar în partea dreaptă o scurtă descriere și un buton care ne deschide fișierul pdf al cărții. Celelalte informații despre carte sunt vizibile în partea de jos a paginii.

Cele două bannere din pagină reprezintă un spațiu pentru anunțuri publicitare pe care administratorul aplicației le poate adăuga ulterior.

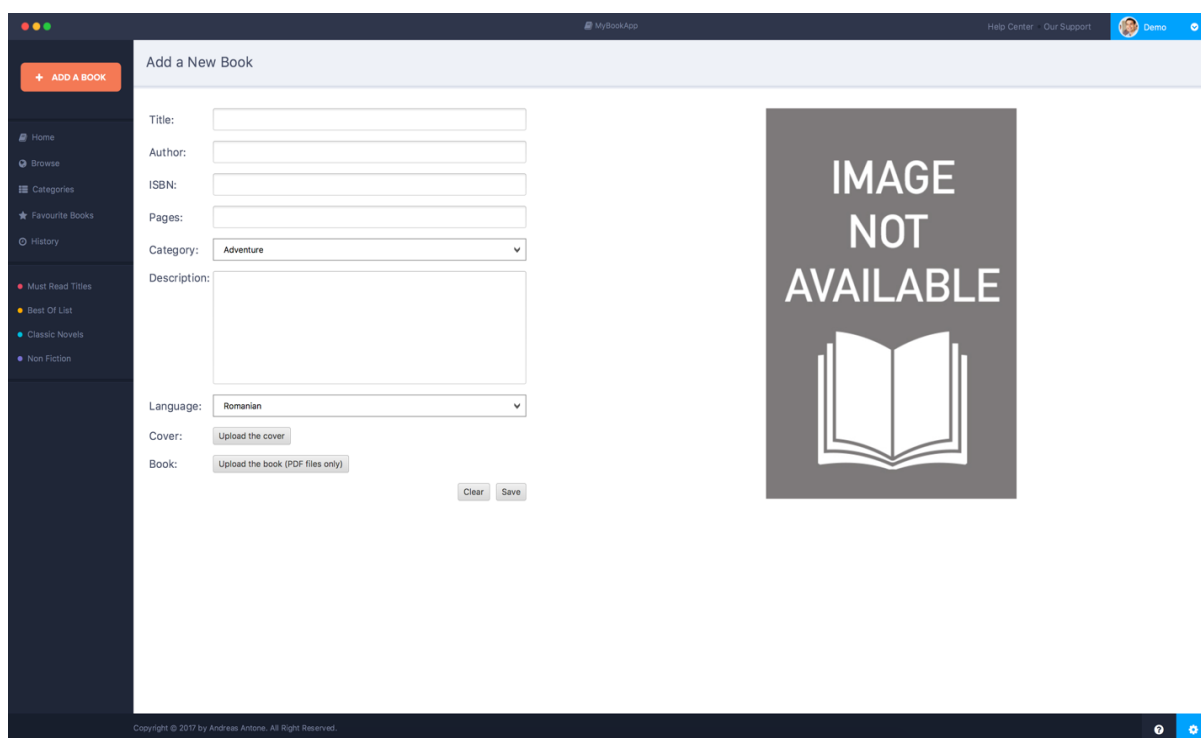


Imaginea 3.2.9 - Pagina de prezentare a unei cărți atunci când cartea este adăugată la favorite



Imaginea 3.2.10 - Pagina de prezentare a unei cărți atunci când cartea nu este adăugată la favorite

În imaginea 3.2.11 putem observa pagina de adăugare a unei cărți. Această pagină este bine gândită și structurată, conținând toate câmpurile necesare pentru identificarea cât mai rapidă a unei cărți.

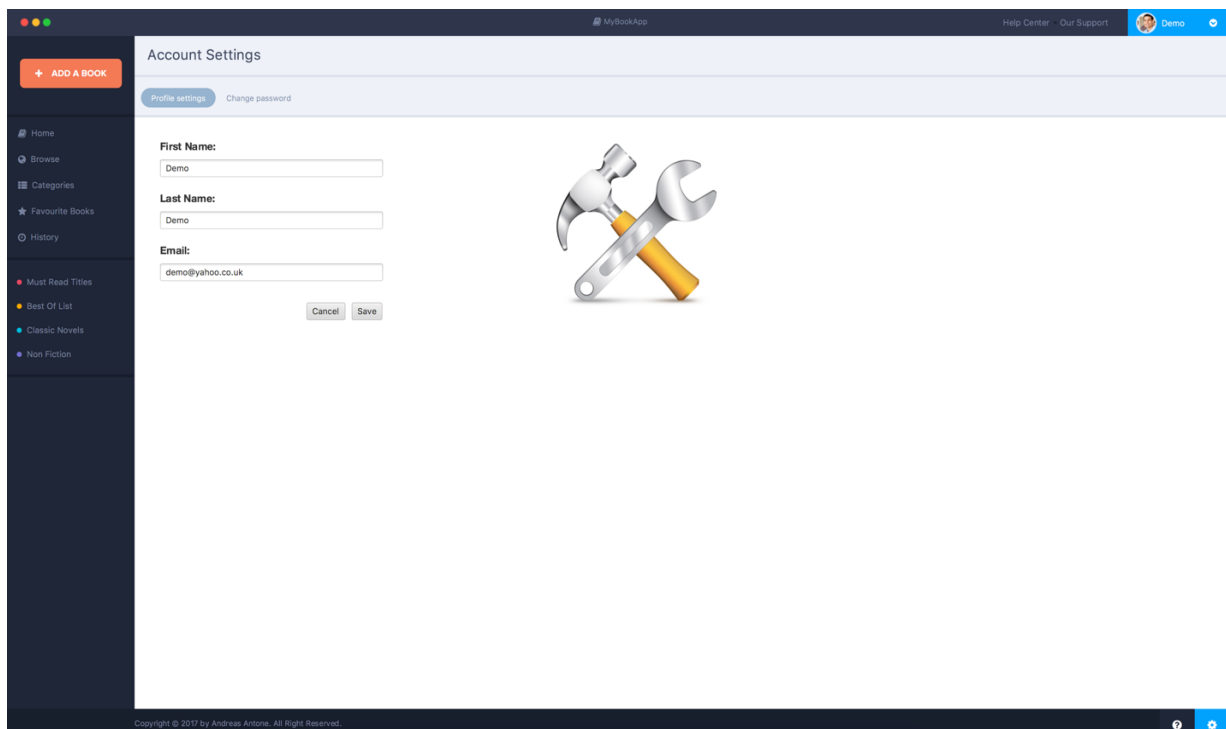


Imaginea 3.2.11 - Pagina de adăugare a unei cărți

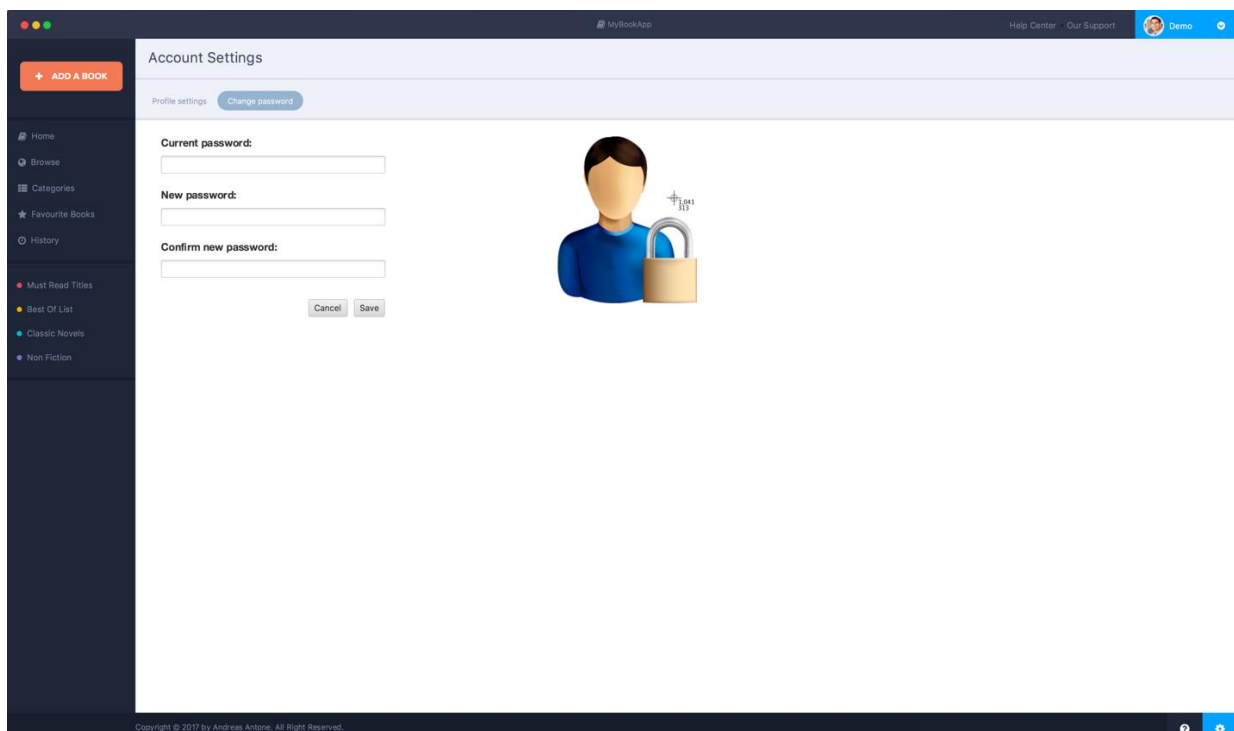
Pagina de setări poate fi accesată rapid prin acționarea butonului de setări simbolizat prin pictograma roțiță și localizat în colțul din dreapta-jos al aplicației. Cu ajutorul acestei pagini utilizatorul își poate schimba parola sau actualiza informațiile contului. Pagina de setări este compusă din mai multe meniuri, însă utilizatorul are la dispoziție doar meniul pentru actualizarea datelor și schimbarea parolei.

Meniul pentru actualizarea informațiilor este prezentat în imaginea 3.2.12 și este alcătuit din câmpurile prenume, nume și email. Toate cele trei câmpuri sunt auto-completate cu informațiile utilizatorului autentificat în program.

Meniul pentru schimbarea parolei poate fi observat în imaginea 3.2.13 și este alcătuit din câmpurile parolă curentă, noua parolă și confirmarea noii parole. Parola nu poate fi schimbată fără a se introduce parola curentă și fără confirmarea noii parole.

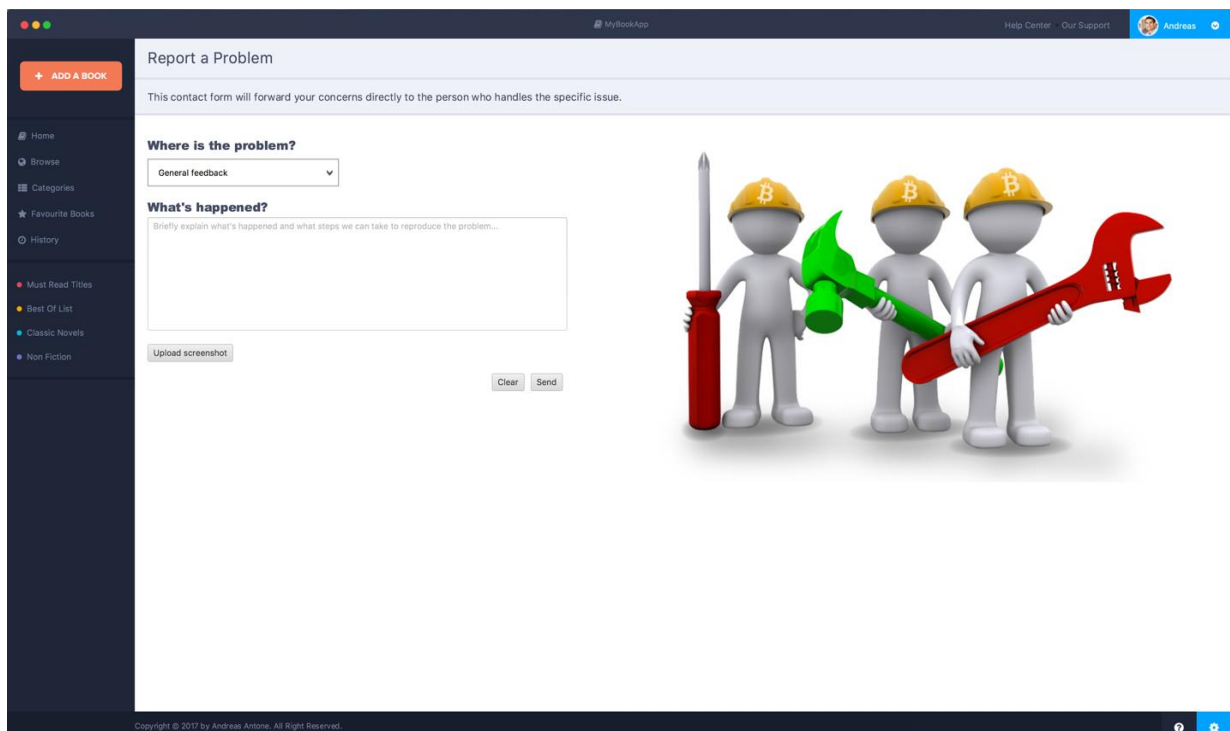


Imaginea 3.2.12 - Pagina de setări ale contului



Imaginea 3.2.13 - Pagina de schimbare a parolei

Pagina de raportare este detaliată în imaginea 3.2.14 și îi permite utilizatorului să raporteze diferite probleme ale aplicației sau să sugereze diverse îmbunătățiri ale acesteia. Această pagină este compusă din subiectul raportării, descrierea acesteia și opțional o captură de ecran.



Imaginea 3.2.14 - Pagina de raportare

CAPITOLUL IV. IMPLEMENTARE

4.1. Pachete și clase

Aplicația este alcătuită din 13 pachete. Fiecare pachet conține clase de obiecte sau resurse ale aplicației, împreună modelând o parte bine delimitată din logica aplicației.

Aplicația conține următoarele pachete:

- ro.unibuc.MyLibrary.main
- ro.unibuc.MyLibrary.main.java
- ro.unibuc.MyLibrary.main.java.common
- ro.unibuc.MyLibrary.main.java.controller
- ro.unibuc.MyLibrary.main.java.database
- ro.unibuc.MyLibrary.main.java.model
- ro.unibuc.MyLibrary.main.java.ui
- ro.unibuc.MyLibrary.main.java.utils
- ro.unibuc.MyLibrary.main.java.view
- ro.unibuc.MyLibrary.main.resources
- ro.unibuc.MyLibrary.main.resources.css
- ro.unibuc.MyLibrary.main.resources.fonts
- ro.unibuc.MyLibrary.main.resources.images

În continuare vor fi detaliate structura și funcționalitatea fiecărui pachet.

Pachetul ro.unibuc.MyLibrary.main conține cele două pachete principale care alcătuiesc aplicația: ro.unibuc.MyLibrary.main.java și ro.unibuc.MyLibrary.main.resources, respectiv pachetul cu clasele Java și pachetul cu resursele aplicației.

- **Pachetul ro.unibuc.MyLibrary.main.java:**

Acest pachet conține la rândul său alte șapte pachete în care sunt stocate fișierele de clase și design-ul aplicației. Pachetul conține o singură clasă numită "Main" cu ajutorul căreia se lansează aplicația.

✓ **Main** este o clasă în care se setează proprietățile aplicației și modul său de rulare. Printre proprietăți putem enumera versiunea aplicației, numele și titlul ferestrei aplicației, tipul ferestrei, iconița aplicației, conexiunea cu baza de date, elemente statice care sunt distribuite în întreaga aplicație etc.

Acest pachet principal conține următoarele pachetele și clasele:

✓ **Pachetul ro.unibuc.MyLibrary.main.java.common:**

Pachetul conține clase prezente în întreaga aplicație, necesare rulării acesteia, dând naștere arhitecturii și elementelor de design. Clasele din acest pachet sunt distribuite cu toate celelalte clase ale aplicației. Pachetul conține următoarele clase:

- **StageManager**: este o clasă care modelează aspectului aplicației. Rolul acestei clase este de a încărca fișierele de design și de a seta dimensiunile ferestrei în funcție de rezoluția ecranului și dimensiunile interfeței grafice. Această clasă conține diverse metode statice, o metodă importantă fiind `setRoot(Parent root)` având ca scop setarea interfeței grafice în funcție de scenariu.

- **ScreenController**: conține o enumerare cu toate ecranele grafice ale aplicației și are ca scop setarea interfeței grafice. Aceasta mai conține o listă cu toate ecranele deschise, precum și o metodă numită `onBack()` care are ca scop întoarcerea la ecranul anterior.

- **Shared**: are ca scop distribuirea unor informații cu toate elementele programului. Această clasă nu conține nicio metodă și are definite doar variabile statice precum: versiunea aplicației, ecranul curent, numele utilizatorului și tipul acestuia (admin sau user), statusul sesiunii curente ș.a.m.d.

✓ **Pachetul ro.unibuc.MyLibrary.main.java.controller**

Pachetul conține clase care au scopul de controlare, adică gestionează comunicarea dintre clasele de tip model și cele de tip view. Așa cum am menționat în capitolele anterioare, aplicația este structurată folosind modelul MVC (engleză Model-View-Controller). Toate clasele din acest pachet conțin un constructor specific interfețelor JavaFX denumit `initialize()`, având ca scop inițializarea diferitelor elemente grafice din interfața prezentată utilizatorului.

Pachetul conține următoarele clase:

- **AddABookController** este clasa care modelează interfața grafică în care utilizatorul poate adăuga diferite cărți și materiale în format pdf. O metodă importantă a acestei clase este `addBook()` care are ca scop extragerea informațiilor din câmpurile completate de utilizator, verifică corectitudinea acestora și trimite informațiile încărcate de utilizator către baza de date și serverul ftp numai dacă acestea îndeplinesc criteriile stabilite de programator. În caz contrar, pe ecranul computerului va fi afișată o fereastră de tipul alert care atenționează utilizatorul că nu îndeplinește anumite criterii.

- **LoginController** este o clasă destinată interfeței inițiale, compusă din constructorul grafic care stabilește dimensiunile ferestrei de logare, titlul acesteia, precum și funcționalitățile butoanelor de închidere, minimizare și maximizare ale ferestrei.

- **SignupController** este una din clasele de bază ale acestui program și are ca scop introducerea interfeței pentru înregistrarea unui cont nou. Deși nu este ecranul inițial al aplicației, această clasă este prima cu care va avea contact un utilizator nou. La fel ca și cele două clase prezentate anterior, în această clasă vom găsi două metode care ne conduc la pagina de logare, respectiv de recuperare a parolei. Metoda `signup()` este metoda de bază a acestei clase și verifică dacă informațiile introduse de utilizator respectă standardele impuse de programator.
- **BrowseController** introduce pagina unde utilizatorul poate să răsfoiască diferite cărți și articole. Scopul acestei clase este de a extrage toate cărțile din baza de date, de a le grupa după anumite criterii și de a le prezenta utilizatorului sub formă de interfață grafică, creând astfel un mediu interactiv pentru acesta. Această clasă conține diferite funcții precum funcția de căutare, sortare și afișare.
- **FavouriteBooksController** este clasa ce modelează pagina cărților favorite ale utilizatorului conectat în aplicație. Toate cărțile marcate cu o stelută de către un utilizator vor fi afișate de această clasă, iar dacă utilizatorul nu are o carte favorită va apărea un mesaj care specifică acest lucru. Clasa este compusă din constructorul grafic care creează panourile și elementele necesare paginii cărților favorite și o metodă cu care sunt extrase și afișate cărțile preferate.
- **HistoryController** are ca scop afișarea unei interfețe în care pot fi găsite toate cărțile accesate de către un utilizator la un moment dat. Aplicația salvează într-o bază de date id-urile cărților deschise de utilizator împreună cu id-ul acestuia. Clasa conține o metodă care extrage din baza de date informațiile cărților deschise de către un utilizator și le modelează într-o interfață grafică inițiată de către constructorul grafic.
- **CategoriesController** are ca și scop sortarea cărților în funcție de categoria în care acestea se încadrează și modelarea unei interfețe grafice alcătuită dintr-un meniu cu categorii și un panou de afișare al cărților. Datele sunt extrase sub forma unei liste din baza de date în funcție de categoria selectată din meniu, iar informațiile listei sunt modelate în obiecte grafice.
- **ViewBookController** este clasa care modelează pagina de afișare a unei cărți selectate și are ca și scop detalierea cărții selectate. Clasa este formată din constructorul grafic care preia informațiile cărții selectate din baza de date în funcție de id-ul acesteia și procesează informațiile în componente grafice creând astfel o interfață prietenoasă cu utilizatorul. În această clasă mai avem metoda `togglefav()` cu ajutorul căreia adăugăm sau ștergem o carte din lista cărților favorite.

- **SettingsController** construiește interfața grafică a panoului de setări care este alcătuit din mai multe interfețe interactive. Utilitatea acestei clase apare din necesitatea utilizatorului de a actualiza anumite informații cu caracter personal. Clasa conține diverse metode precum metoda de schimbare a parolei sau de actualizare a adresei email.
- **LoginMessageController** aparține interfeței inițiale și are ca scop afișarea unor mesaje de confirmare, atenționare sau informare în interfața inițială. Este alcătuită din constructorul grafic și o metodă care întoarce utilizatorul la ecranul anterior. Mesajul și tipul acestuia sunt setate în constructorul grafic, acestea fiind stocate în clasa **Shared** despre care am vorbit mai sus.
- **HomeController** este clasa care conține primul ecran din interfața principală și anume pagina de start. Clasa conține numai constructorul grafic care inițializează pagina de start și extrage din baza de date o maximă sau un citat în funcție de ziua curentă.
- **RecoverController** are ca scop introducerea interfeței pentru recuperarea parolei și a elementelor ce alcătuiesc interfața. Această clasă conține o metodă cu ajutorul căreia ne putem întoarce la interfața inițială, adică la pagina de logare, o metodă cu care putem accesa pagina de înregistrare precum și metoda de bază a acestei clase denumită `recover()`. Metoda `recover()` verifică dacă email-ul introdus în căsuța email este într-un format corespunzător, dacă corespunde unui utilizator din baza de date, iar dacă aceste criterii au fost îndeplinite trimite un email de resetare a parolei.
- **ApplicationController** este clasa interfeței principale și are scopul de a încărca elementele care alcătuiesc interfața principală și de a stabili dimensiunile acesteia. În această clasă vom găsi diferite metode care ne conduc pe diferite pagini din aplicației, precum și constructorul grafic care inițializează interfața și se ocupă cu reîmprospătarea acesteia în urma operațiilor executate de utilizator.
- **HelpController** a fost concepută pentru a oferi sprijin și suport utilizatorului, dar și pentru îmbunătățirea aplicației prin eliminarea unor erori raportate de către utilizatori sau implementarea unor sugestii trimise de aceștia. Cu ajutorul metodei `openScreenshot()`, utilizatorul poate atașa capturi din aplicație sau mesaje de eroare apărute în timpul utilizării acesteia, ajutând astfel programatorul să remedieze problemele într-un timp cât mai scurt.
- **SigninController** este prima clasă cu care are contact utilizatorul, deoarece este prima interfață apărută pe ecranul calculatorului în momentul rulării aplicației. Scopul clasei este de a inițializa interfața grafică specifică paginii de logare, precum și proprietățile acesteia. Clasa conține două metode care au ca scop schimbarea interfeței către pagina de recuperare a parolei și pagina de înregistrare. Metoda de bază a acestei clase este metoda `login()` care verifică dacă informațiile introduse în câmpul utilizator și parolă sunt corecte, respectiv dacă acestea

îndeplinesc următoarele criterii: câmpul pentru utilizator și parolă sa nu fie vide, utilizatorul să fie compus din cel puțin 3 caractere și parola sa aibă minimum 6 caractere, iar informațiile introduse să fie specifice unui utilizator memorat în baza de date.

✓ **Pachetul ro.unibuc.MyLibrary.main.java.view**

Pachetul conține toate fișierele layout necesare pentru modelarea interfețelor grafice ale aplicației. Acestea conțin un cod asemănător cu limbajul XML și toate obiectele grafice necesare interfețelor grafice. Fiecare clasă din pachetul de controlere are un fișier de layout corespondent care se află în acest pachet și asigură relația view-controller.

Pachetul conține următoarele layout-uri:

- | | | |
|--------------------|----------------------|------------------|
| • ViewBookView | • FavouriteBooksView | • SettingsView |
| • LoginMessageView | • ApplicationView | • SigninView |
| • BrowseView | • HelpView | • RecoverView |
| • LoginView | • HomeView | • CategoriesView |
| • HistoryView | • SignupView | • AddABookView |

✓ **Pachetul ro.unibuc.MyLibrary.main.java.model**

Pachetul conține clase care au scopul de modelare a informațiilor din aplicație. Clasele din acest pachet se mai numesc și modele, fiind de fapt obiectele principale ale aplicației.

Pachetul conține următoarele clase:

- **User** este clasa care definește obiectul utilizator și are ca scop stocarea informațiilor unui utilizator. Mulțimea de date care definește obiectul utilizator este alcătuită din numele și prenumele utilizatorului, email-ul, parola, user-ul și tipul acestuia, data când contul acestuia a fost creat și ultima dată când a accesat aplicația. Clasa are mai mulți constructori pentru a satisface diferite cazuri.
- **User Book** este o clasă care modelează obiectul carte și conține o serie de constructori utili în diferite scenarii din aplicație. Pe lângă constructori mai întâlnim și get-eri și set-eri pentru fiecare atribut al obiectului carte. Printre proprietățile obiectului carte întâlnim id-ul, numele, autorul, ISBN-ul, numărul de pagini, coperta, categoria, descrierea, limba, data publicării, numărul de vizualizări, adresa web unde este stocată aceasta, precum și id-ul utilizatorului care a adăugat cartea.

- **User Book Report** este o clasă alcătuită dintr-un constructor și metode necesare stocării și citirii obiectului de tip report. Scopul acestei clase este de a defini un obiect folosit pentru raportarea erorilor și al sugestiilor. Obiectul report este alcătuit din următoarea mulțime de date: categorie, descriere, captură de ecran, data raportării și status. Captura de ecran este un atribut opțional, dar foarte util pentru remedierea erorilor întâmpinate de utilizator.
- **Quote** este clasa care definește obiectul citat, fiind compusă dintr-un constructor și mulțimea de date ale acestuia. Obiectul citat este format din textul și autorul citatului.

✓ **Pachetul ro.unibuc.MyLibrary.main.java.database**

Pachetul conține o clasă necesară pentru stabilirea conexiunii cu baza de date și clase folosite la stocarea datelor. Pentru fiecare tip de obiect stocat în baza de date avem câte o clasă corespondentă acestuia.

Pachetul conține următoarele clase:

- **ConnectToDatabase** este clasa cu ajutorul căreia stabilim conexiunea cu baza de date. Această clasă se apelează la rularea programului și folosește librăria externă java.sql. În această clasă întâlnim și setările serverului SQL necesare conectării la baza de date.
- **UsersDatabase** este o clasă care execută diferite operații asupra tabelului din baza de date unde sunt stocați utilizatorii. În această clasă întâlnim diferite metode care au ca scop adăugarea, citirea, actualizarea sau ștergerea unui utilizator. Mai întâlnim și metodele necesare logării, recuperării parolei sau schimbarea acesteia, precum și o metodă care extrage într-o listă toți utilizatorii stocați în baza de date.
- **BooksDatabase** este clasa care are ca scop salvarea cărților într-o bază de date bine definită. Clasa conține diferite metode utile pentru adăugarea cărților în baza de date, pentru citirea sau gruparea acestora într-o listă, căutarea acestora în funcție de anumite cuvinte cheie sau criterii de căutare, precum și metode specifice pentru istoricul utilizatorului și cărțile favorite ale acestuia.
- **ReportsDatabase** scopul acestei clase este de a memora sugestiile utilizatorului și problemele întâmpinate de acesta într-un tabel specific din baza de date.
- **QuotesDatabase** este o clasă compusă dintr-o singură metodă denumită getQuote() care returnează un obiect de tipul **Quote**. Scopul acestei clase este de a extrage un citat din baza de date în funcție de ziua curentă și afișarea acestuia pe pagina principală a aplicației.

✓ **Pachetul ro.unibuc.MyLibrary.main.java.ui**

Pachetul conține o singură clasă numită **AlertMessage** care are ca scop afișarea unor ferestre de dimensiuni mici cu scop informativ. În această clasă avem doar o singură metodă denumită `AlertMessage(String title, String content, AlertType type)` cu trei parametri. Parametrul `title` este un text și reprezintă titlul ferestrei, `content` este un text informativ afișat în interiorul ferestrei, iar `type` este tipul mesajului (confirmare, eroare, informare sau avertizare).

✓ **Pachetul ro.unibuc.MyLibrary.main.java.utils**

Acest pachet are un nume sugestiv și conține toate clasele utile aplicației la un moment dat. Clasele folosite în acest pachet nu sunt folosite în permanență, iar metodele acestora sunt folosite de mai multe clase din program.

Pachetul conține următoarele clase:

- **ValidateEmail** este o clasă compusă dintr-o singură metodă de tipul boolean denumită `validate(String email)` și o variabilă constantă. În variabilă s-a memorat o informație de tipul text, reprezentând șablonul unui email, iar în metodă s-a folosit un pattern pentru a verifica dacă parametrul primit îndeplinește criteriile șablonului.
- **SendEmail** este o clasă care folosește librăria externă [javax.mail](#) și are ca scop trimiterea unor email-uri automate. Pentru folosirea librăriei externe avem nevoie de un server și de o adresă email. În această clasă întâlnim o singură metodă denumită `send (String to, String subject, String content)`, unde "to" este destinatarul email-ului, "subject" este subiectul email-ului, iar "content" este conținutul acestuia. Trebuie menționat faptul că toate setările librăriei se găsesc în metoda **send**. Metoda este des întâlnită la recuperarea parolei, atunci când se trimite un email cu o nouă parolă.
- **OpenURL** este o clasă compusă din metoda `open (String url)` și primește ca argument o adresă web, având ca scop deschiderea adresei web în browserul prestabilit. Metoda `open` verifică sistemul de operare al calculatorului și în funcție de acesta apelează comanda de control împreună cu sintaxa de deschidere a unei adrese de internet.
- **ConvertDate** este o clasă care conține o metodă denumită `convert (Date dt)` și are ca scop transformarea unei date calendaristice într-o valoare întreagă. Metoda primește ca argument o dată calendaristică și o transformă în secunde, minute, ore sau zile.
- **FTPUploader** este una din clasele de bază ale acestui proiect și are ca scop încărcarea cărților și a copertilor pe un server web. Această clasă folosește librăria externă [org.apache.commons.net](#) pentru a stabili o conexiune ftp client-server, unde client este

utilizatorul și server este locația web unde sunt încărcate fișierele. În această clasă vom întâlni o singură metodă denumită FTPUploader (String host, String user, String pwd) care primește trei parametri de tipul text și stabilește conexiunea cu serverul ftp. Primul parametru reprezintă adresa ftp a serverului, iar al doilea și al treilea parametru reprezintă utilizatorul și parola necesare pentru autentificarea la serverul ftp.

✓ Pachetul ro.unibuc.MyLibrary.main.resources

Acest pachet conține resursele propriu-zise ale aplicației, care au rolul de a da un aspect plăcut interfețelor grafice și de a le face mai prietenoase. Printre acestea se numără fișierele de tip font, fișierele css pentru stilizare și diverse imagini grafice.

4.2. Baza de date

Baza de date a aplicației este formată din schema de date numită ”mybookapp” și cele 6 tabele de date conținute de aceasta. Asupra bazei de date putem executa operațiunile de adăugare, actualizare, interogare și ștergere a datelor.

Schema conține următoarele tabele:

- BOOKS
- FAVORITE_BOOKS
- HISTORY
- QUOTES
- REPORT
- USERS

Tabelul **BOOKS** este tabelul în care sunt stocate informațiile despre cărțile din aplicație. Tabelul este alcătuit din următoarele câmpuri:

- idBOOK - Este un id unic, având rolul de cheie primară. Acesta nu poate fi nul și este auto-incrementat.
- title - Este un șir de maximum 125 de caractere și stochează titlul cărții. Această informație nu poate fi nulă.
- author - Este un șir de maximum 125 de caractere și stochează autorul cărții. Această informație nu poate fi nulă.
- ISBN - Stochează valoarea ISBN corespunzătoare fiecărei cărți și poate avea o lungime maximă de 50 de caractere. Această informație nu poate fi nulă.
- pages - Stochează o valoare întreagă și reprezintă numărul de pagini pe care le conține o carte. Această informație nu poate fi nulă.

- cover - Stochează adresa web unde poate fi găsită coperta cărții și nu poate să depășească numărul de 145 de caractere. Această informație nu este obligatorie, deoarece are o valoare prestabilită.
- description - Este un șir de 5000 de caractere și reprezintă descrierea cărții. Acest field nu este obligatoriu.
- language - Este un șir de 65 de caractere și stochează limba cărții. Informația stocată nu poate fi nulă.
- link - Este un șir de 145 de caractere și reprezintă adresa web unde este stocată cartea. Acest field nu poate fi nul.
- userID - Este o valoare numerică și stochează id-ul utilizatorului care a adăugat cartea. Această informație nu poate să lipsească.
- views - Este o valoare numerică care crește de fiecare dată când o carte este accesată de către un utilizator și reprezintă numărul de vizualizări al unei cărți. Informația poate să lipsească, deoarece are o valoare prestabilită egală cu 0.
- publication_date - Stochează o dată calendaristică și reprezintă data la care a fost adăugată cartea în baza de date. Această informație nu poate să lipsească.
- approved - Stochează o valoare logică și reprezintă starea unei cărți. Cărțile recent adăugate au valoarea prestabilită egală cu 0, deoarece sunt în curs de prelucrare și trebuiesc aprobate de către un administrator sau moderator, moment în care valoare devine 1.

Tabelul **HISTORY** stochează id-urile cărților accesate de către un utilizator, precum și id-ul acestuia. Tabelul este alcătuit din următoarele field-uri:

- id - Stochează o valoare întreagă cu rol de cheie primară. Aceasta se auto-incrementează și nu poate fi vidă.
- idUser - Stochează id-ului utilizatorului care adaugă cartea la favorite și are rol de cheie externă. Valoarea stocată este una de tipul numeric și nu poate să lipsească.
- idBook - Stochează id-ul cărții adăugate la favorite de către utilizator și are rol de cheie externă. Valoarea stocată este una de tipul numeric și nu poate să lipsească.
- time - Stochează o dată calendaristică și reprezintă data la care o carte a fost adăugată la favorite. Informația nu poate să lipsească.

Tabelul **FAVORITE_BOOKS** stochează id-urile cărților adăugate la favorite de către utilizatori, precum și id-ul utilizatorului. Tabelul este alcătuit din următoarele field-uri:

- **idHISTORY** - Stochează o valoare întreagă cu rol de cheie primară. Aceasta se auto-incrementează și nu poate fi vidă.

- **idUser** - Stochează id-ul utilizatorului care a accesat o carte și are rol de cheie externă. Valoarea stocată este una de tipul numeric și nu poate să lipsească.

- **idBook** - Stochează id-ul cărții vizualizate de către utilizator și are rol de cheie externă. Valoarea stocată este una de tipul numeric și nu poate să lipsească.

Tabelul **QUOTES** stochează o serie de 31 de maxime și citate, precum și autorul acestora.

Tabelul este alcătuit din următoarele field-uri:

- **id** - Stochează o valoare întreagă cu rol de cheie primară. Aceasta se auto-incrementează și nu poate fi vidă.

- **quote** - Este un șir de maximum 500 de caractere și stochează maxima sau citatul. Informația nu poate fi nulă.

- **author** - Este un șir de 65 de caractere și reprezintă autorul maximei sau al citatului. Informația este obligatorie.

Tabelul **REPORT** stochează problemele raportate de utilizatori, precum și sugestiile acestora. Tabelul este alcătuit din următoarele field-uri:

- **id** - Stochează o valoare întreagă cu rol de cheie primară. Aceasta se auto-incrementează și nu poate fi vidă.

- **category** - Memorează un șir de maximum 45 de caractere și reprezintă tipul raportării. Categoria poate fi: feedback general, probleme ale aplicației și conținut abuziv. Acest câmp nu poate fi nul.

- **description** - Stochează un șir de maximum 500 de caractere și are ca scop detalierea problemei întâmpinate sau a sugestiilor propuse. Informația este obligatorie.

- **screenshot** - Memorează un șir de maximum 100 de caractere și reprezintă adresa web unde este stocată captura cu eroarea aplicației. Această informație poate să lipsească.

- **reportDate** - Stochează o dată calendaristică și reprezintă data la care o raportare a fost adăugată în baza de date. Informația nu poate să lipsească.

- **status** - Stochează o valoare logică și reprezintă starea unei raportări. Raportările în curs de procesare au valoarea implicită 0, iar cele finalizate valoarea 1.

Tabelul **USERS** stochează informațiile cu caracter personal ale utilizatorilor și informații specifice despre aceștia. Tabelul este alcătuit din următoarele field-uri:

- ID - Stochează o valoare întreagă cu rol de cheie primară. Aceasta se auto-incrementează și nu poate fi vidă.
- USERNAME - Memorează un șir de maximum 45 caractere și reprezintă pseudonimul utilizatorului. Informația nu poate să fie nulă.
- EMAIL - Memorează un șir de maximum 45 caractere și reprezintă adresa de email a utilizatorului. Acest câmp este obligatoriu.
- PASSWORD - Memorează un șir de maximum 45 caractere, reprezentând parola utilizatorului. Informația stocată este codată în MD5 și nu poate să lipsească.
- LAST_NAME - Memorează un șir de maximum 45 caractere și reprezintă numele utilizatorului. Informația nu poate să lipsească.
- FIRST_NAME - Memorează un șir de maximum 45 caractere și reprezintă prenumele utilizatorului. Informația nu poate să lipsească.
- TYPE - Stochează un șir cu lungimea maximă de 15 caractere și reprezintă tipul utilizatorului. Tipul utilizatorului poate fi user, admin sau moderator. Câmpul nu este obligatoriu, deoarece are valoarea implicită user.
- JOIN_DATE - Memorează o dată calendaristică, reprezentând data la care contul a fost creat. Această informație este completată automat în momentul adăugării unui cont nou în baza de date și nu poate fi nulă.
- LAST_LOGIN - Memorează o dată calendaristică și reprezintă ultima dată când utilizatorul s-a conectat în program. În momentul adăugării unui cont nou, acest câmp este nul, deci informația poate să lipsească până la prima logare în aplicație.

CONCLUZII

În lucrarea de față am analizat subiectul bibliotecilor digitale și scopul acestora în secolul XXI. Rezultatele obținute arată necesitatea acestora în viața cotidiană, precum și utilitatea acestora.

În primul rând, ca aplicația să aibă succes și să funcționeze este nevoie de un moderator sau administrator care să urmărească permanent cărțile introduse în baza de date, problemele întâmpinate de utilizatori și sugestiile acestora. Utilizatorii aplicației joacă și ei un rol important, deoarece ei sunt cei care adăugă diferite cărți și materiale, iar fără ei dezvoltarea aplicației nu ar fi posibilă.

Deși am petrecut o cantitate semnificativă de timp pentru a dezvolta aplicația, consider că există câteva neajunsuri în aplicabilitatea ei practică.

În timpul rulării, aplicația consumă o cantitate importantă de memorie, iar adăugarea de thread-uri ar contribui la o viteză de rulare mai bună. Folosirea în mod constant a aplicației, precum și sugestiile primite de la utilizatori pot ajuta la eliminarea liniilor de cod redundante din aplicației, precum și a paginilor care nu își găsesc utilitatea, făcându-se astfel o economie semnificativă de cod. O altă ajustare ar fi adăugarea serviciilor web sau folosirea unei baze de date în Cloud.

În concluzie, lucrarea de față reprezintă un punct de plecare important în dezvoltarea bibliotecilor digitale, oferind o serie de observații și implementări în acest sens. Aplicația Java dezvoltată în scot academic și demonstrativ reprezintă un punct bun de plecare al bibliotecilor online și poate fi folosită și îmbunătățită pe viitor.

Din punctul meu de vedere, utilitatea aplicației este indiscutabilă. Crearea unor variante mai complexe și mai stabilite ale aplicației poate facilita legătura dintre om, literatură și computer.

Consider că aplicația merită să fie îmbunătățită în viitor, iar folosirea acesteia într-o instituție de învățământ poate reprezenta un punct de plecare.

BIBLIOGRAFIE

1. Curs practic de Java, Cristian Frăsinaru, Editura Matrixrom, 2005
2. Fundamente Java, Eugen Petac și Tudor Udrescu, Editura Matrixrom, 2005
3. JavaFX For Dummies, Doug Lowe, Editura Wiley, 2014
4. Tehnologii XML - XML in JAVA - XML pentru avansați, L. Anghel, Editura Albastra, 2007
5. Baze de date. Aplicații ORACLE - SQL și PL/SQL pentru elevii claselor a XII-a, matematică-informatică, Doina Narcisa Merlan, Editura Else, 2011
6. <http://gta.math.unibuc.ro/pages/Algebra1TD.pdf>
7. https://www.tutorialspoint.com/swift/swift_tutorial.pdf
8. <https://carlosicaza.com/swiftbooks/SwiftLanguage.pdf>
9. <http://onlinelibrary.london.ac.uk/about>
10. <https://en.wikipedia.org/wiki/Scribd>
11. https://ro.wikipedia.org/wiki/Bibliotec%C4%83_digital%C4%83
12. <http://www.open.ac.uk/library/library-videos/what-is-the-online-library>
13. <http://ebooks.unibuc.ro/StiinteCOM/bibliologie/6.htm>
14. <http://www.open.ac.uk/library/news/what-is-an-online-library-and-why-use-it>
15. <http://radicalreference.info/book/export/html/1255>
16. <http://www.navarrocollege.edu/library/about/ten-good-reasons/>
17. <https://www.librariesareessential.com/why-are-libraries-essential/>
18. <https://blog.mozilla.org/internetcitizen/2017/09/04/libraries/>
19. <http://lesk.com/mlesk/follett/follett.html>
20. <http://lnu.diva-portal.org/smash/get/diva2:1129431/FULLTEXT01.pdf>
21. <https://pdfs.semanticscholar.org/d0e6/90b74b3b9513d9d1f97cf366e31a3920a4bf.pdf>
22. <http://www.lisr.ro/7-tirziman2.pdf>
23. <https://www.infozoom.ro/articole/avantajele-si-dezavantajele-cartilor-electronice/>
24. <http://evaluare.fmi.unibuc.ro/>
25. <https://github.com/cathive/fonts-fontawesome>
26. <https://github.com/Haixing-Hu/javafx-widgets>
27. <http://dribbble.com/trendyWebStar>
28. <https://www.behance.net/gallery/14643125/Donate-A-Book>
29. <http://www.tutorialspoint.com/java/>
30. <http://www.tutorialspoint.com/javafx/>
31. <https://docs.oracle.com/javafx/2/overview/jfxpub-overview.htm>
32. <https://commons.apache.org/proper/commons-net/>
33. <http://www.oracle.com/technetwork/java/javamail/index.html>
34. <https://dev.mysql.com/downloads/connector/j/8.0.html>