

Ministerul Educației și Cercetării al Republicii Moldova
Universitatea Tehnică a Moldovei
Facultatea Calculatoare, Informatică și Microelectronică
Departamentul Inginerie Software

Admis la susținere

Șef departament:

Nume, Prenume, titlul științifico-didactic, titlul științific

„_____” _____2023

**„ Platformă de gestionare a testelor de evaluare sumativă și periodică pentru disciplinele școlare cu
profil real ”**

Teză de licență

Student: Pavalachi Alina, Grupa TI-194

Coordonator: Bulai Rodica, lector universitar

Consultant:

Consultant:

Recenzat:

Chișinău, 2023

MINISTERUL EDUCAȚIEI ȘI CERCETĂRII AL REPUBLICII MOLDOVA

UNIVERSITATEA TEHNICĂ A MOLDOVEI

CAIETUL

STAGIULUI DE PRACTICĂ

PENTRU STUDENȚII STUDIILOR SUPERIOARE DE LICENȚĂ – CICLULUI I

Stagiul de practică _____ *de licență* _____
tipul stagiului de practică

A studentului (ei) _____ *Pavalachi Alina* _____
numele, prenumele

Facultatea _____ *Calculatoare, Informatică și Microelectronică* _____

Programul de studii _____ *Tehnologia informației* _____

Ciclul _____ *I* _____ Anul de studii _____ *IV* _____ Grupa _____ *TI-194* _____

Locul stagiului de practică _____ Codwer _____
denumirea unității economice

Conducătorul stagiului de practică
de la UTM _____ *asist.univ. Cojocaru Svetlana* _____
funcția, numele, prenumele

Conducătorul stagiului de practică
de la unitatea economică _____ *Popescu Mihai* _____
funcția, numele, prenumele

Chișinău, 2023

I. CAIETUL DE SARCINI

| Nr. crt. | Tematica lucrărilor preconizate | Termene planificate | |
|----------|---|---------------------------|-----------------|
| | | început/ sfârșit | numărul de zile |
| 1 | <i>Analiza domeniului de studii și Modelarea și proiectarea sistemul informatic</i> | 30.01.2023- 03.02.2023 | 5 zile |
| 2 | <i>Realizarea sistemului</i> | 06.02.2023- 10.02.2023 | 5 zile |
| 3 | <i>Realizarea sistemului</i> | 20.02.2023- 24.02.2023 | 5 zile |
| 4 | <i>Realizarea raportului și prezentării</i> | 27.02.2023- 03.03.2023 | 5 zile |

II. SARCINA INDIVIDUALĂ (SE INDICĂ TEMA PROIECTULUI DE LICENȚĂ)

Tema: Platformă de gestiune a testelor de evaluare pentru discipline cu profil real (Fizică, Matematică)

| Nr. crt. | Conținutul concis al sarcinilor individuale | Mențiune cu privire la realizare |
|----------|---|--|
| 1 | <i>Analiza domeniului de studii</i> | <ul style="list-style-type: none"> • Se descrie domeniul din care face parte proiectul creat (tehnologii informaționale). Ce tip de aplicație (web, mobile...etc). În ce domeniu se va aplica proiectul (învățământ, sfera serviciilor.....etc). • Se va argumenta importanța creării unui astfel de proiect. • Se va face o descriere a cel puțin 3 sisteme deja existente (sau sisteme care au ceva comun din punct de vedere funcțional cu ce se va face în proiect). Se va face o comparare a sistemelor descrise. • În baza comparării se va scri scopul și obiectivele, cerințele sistemului. Determinarea cerințelor funcționale, nefuncționale. Determinarea funcționalului sistemului și subsistemelor. Se va descrie cât mai detaliat caietul de sarcini a proiectului. |
| 2 | <i>Modelarea și proiectarea sistemul informatic</i> | <ul style="list-style-type: none"> • Descrierea comportamentală a sistemului (imaginea generală asupra sistemului; modelarea vizuală a fluxurilor; stările de tranzație a sistemului (Statechart Diagram); descrierea scenariilor de utilizare a aplicației (Sequence Diagram); fluxurile de mesaje și legăturile dintre componentele sistemului (Collaboration Diagram). • Descrierea structurală a sistemului (descrierea structurii statice a sistemului; relațiile de dependență între componentele; modelarea echipamentelor mediului de implementare. |
| 3 | <i>Realizarea sistemului</i> | <ul style="list-style-type: none"> • Se face o introducere în ce s-a folosit pentru realizarea sarcinilor. Se descrie pe scurt tehnologiile folosite, limbaje de programare utilizate, instrumente, algoritmi, șabloane...etc , totul ce s-a folosit la realizarea proiectului. • Se descrie la nivel de cod realizarea funcționalităților • Se descrie testarea sistemului |
| 4 | <i>Realizarea raportului</i> | <ul style="list-style-type: none"> • De structurat după sarcini individuale • De redactat confor standardelor de redactare |
| 5 | <i>Realizarea prezentării</i> | <ul style="list-style-type: none"> • Slide cu denumirea temei • Slide cu Analiza domeniului • Slide cu Scopul și Obiectivele proiectului • Slide cu Imaginea generală a sistemului • Slide cu Descrierea structurii statice a sistemului • Slide cu Instrumente, tehnologii utilizate pentru realizarea sistemului • Slide cu Video- prezentarea aplicației |

Conducătorul _____ / Cojocaru Svetlana /
semnătura numele și prenumele

DECLARAȚIA STUDENTULUI

Subsemnata Pavalachi Alina, declar pe proprie răspundere că lucrarea de față este rezultatul muncii mele, pe baza propriilor cercetări și pe baza informațiilor obținute din surse care au fost citate și indicate, conform normelor etice, în note și în bibliografie. Declar că lucrarea nu a mai fost prezentată sub această formă la nici o instituție de învățământ superior în vederea obținerii unui grad sau titlu științific ori didactic.

Pavalachi Alina

REZUMAT

Pavalachi Alina, teză de licență cu tema: „ Platformă pentru gestionarea testelor de evaluare sumativă pentru discipline cu profil real”

Lucrarea respectivă este divizată în 5 capitole ce încapsulează informația despre produsul în sine și despre realizarea acestuia, și anume:

- analiza domeniului de studiu, în care se regăsește informația despre actualitatea temei alese, obiectivele și scopul pe care se centrează acest produs, precum și alte informații despre produsele similare acestuia, cerințele față de acest produs și caietul de sarcini;
- modelarea și proiectarea sistemului informatic, încorporează toate diagramele UML urmate de descrierea și explicațiile acestora. Este divizat în două subcapitole, primul ilustrând caracteristicile sistemului din punct de vedere comportamental cu ajutorul diagramelor cazurilor de utilizare, a diagramelor de activități și a diagramelor de secvență, iar al doilea din punct de vedere structural prin intermediul diagramelor de componente și a celor de deploy;
- realizarea sistemului, care este divizat în două subcapitole primul ilustrând implementarea funcționalităților de bază, iar al doilea testarea sistemului și verificarea parametrilor calității;
- documentarea produsului, cuprinde specificațiile produsului și instrucțiunile de utilizare a acestuia;
- estimarea costurilor produsului, ilustrând cheltuielile din punct de vedere al timpului, al resurselor și al finanțelor, estimate pentru procesul de elaborare a produsului respectiv.

Cuvinte cheie: Testare, Analiza, Evaluare, Barem, Școală.

Platforma de gestionare a testelor de evaluare sumativă pentru disciplinele cu profil real este orientat către instituțiile de învățământ gimnazial, liceal și posibil și către universități și colegii ce au tangențe cu disciplinele caracteristice profilului real.

Utilizatorii acestui produs vor fi atât profesorii cât și elevii studenții ce își fac studiile în instituțiile de învățământ acreditate din Republica Moldova.

CUPRINS

| | |
|---|-----------|
| INTRODUCERE..... | 8 |
| 1 ANALIZA DOMENIULUI DE STUDIU..... | 10 |
| 1.1 Analiza soluțiilor existente și compararea acestora | 11 |
| 1.2 Scopul și obiectivele propuse..... | 16 |
| 1.3 Importanța și posibilitățile sistemului | 17 |
| 1.4 Cerințe funcționale și nefuncționale. | 21 |
| 1.5 Caietul de sarcini..... | 25 |
| 2 MODELAREA ȘI PROIECTAREA SISTEMULUI INFORMATIC..... | 27 |
| 2.1 Modelarea comportamentală..... | 28 |
| 2.2 Modelarea structurală..... | 38 |
| 3 REALIZAREA SISTEMULUI. | 44 |
| 3.1 Implementarea funcționalităților de bază..... | 46 |
| 3.2 Testarea sistemului..... | 49 |
| 4 DOCUMENTAREA PRODUSULUI REALIZAT | 52 |
| 5 ESTIMAREA COSTURILOR PROIECTULUI | 54 |
| CONCLUZII..... | 57 |
| BOBLIOGRAFIE | 59 |
| ANEXE..... | 60 |
| 1 Documentul în formă de test fulger | 60 |
| 2 Documentul în formă de test cu nivel avansat de dificultate | 61 |
| 3 Documentul în formă de test cu nivel mediu de dificultate | 62 |

INTRODUCERE

Este evident că în societatea modernă utilizarea Tehnologiilor Informaționale și Comunicabile devine un punct de referință al schimbărilor fundamentale ale sistemului de învățământ iar transformarea digitală poate reprezenta o soluție la provocările actuale ale sistemului educațional din Republica Moldova. Deși, la nivel de guvern există o recomandare clară pentru folosirea de soluții moderne în educație, care recomandă în mod expres educația prin activitatea curriculară și extra curriculară bazată pe TIC, marea majoritate a școlilor se confruntă cu lipsa de fonduri proprii necesare implementării și adopției tehnologiei necesare.

Cadrele didactice, părinții și elevii au nevoie de soluții care să fie accesibile, ușor de folosit și de administrat și care să ofere un conținut educațional vast. Așadar, profesorii, administrația și părinții au înțeles rapid valoarea adăugată a unei soluției complete care face mai accesibilă pregătirea, gestionarea și folosirea mijloacelor educaționale moderne pentru școli. Această soluție are numeroase beneficii principale fiind:

- scăderea numărului de absențe;
- creșterea mediei pe școală;
- centrarea activității profesorilor pe actul educațional, decât pe aspecte administrative;
- vizibilitate crescută asupra activității școlare a copilului, pentru părinți;
- grafic flexibil și posibilitatea de a revizualiza conținutul cursului;
- îmbunătățirea comunicării dintre cadrele didactice și administrația școlii;
- îmbunătățirea comunicării dintre cadrele didactice și elevi;
- acces nelimitat la resursele educative, materialele teoretice precum și cele practice, etc.

Cu toate că importanța digitalizării nu mai stă sub semnul întrebării, rezultatele oferite în urma digitalizării lasă de dorit în mare parte din cauza lipsei de resurse financiare sau din cauza numărului mic al cadrelor didactice ce posedă cunoștințe digitale.

O experiență timidă de utilizare a tehnologiilor informaționale și comunicaționale la lecții a demonstrat că:

- se mărește motivația de învățare a disciplinei de studii;
- apare posibilitatea de a crea produsele colective în sistemul on-line pentru lucru în comun al elevilor și profesorilor;
- se pot prezenta experimente virtuale, care nu pot fi efectuate în condițiile laboratorului din școală;
- se pot prezenta situații concrete și arăta graficele la relație între diferite mărimi ce caracterizează un proces sau altul la examinarea situațiilor concrete, deschise în sarcinile de învățare la o anumită disciplină;
- se pot demonstra circuite electrice în funcție, modele etc;

- se apreciază ușor și rapid asimilarea celor studiate;
- se poate arăta instalarea modelelor, demonstra experimente interactive la una din sarcinile de învățare, de a demonstra modelele unor fenomene;
- se creează posibilitatea pentru elevi de a elabora în comun cu profesorul demersul procesului didactic;
- permite elevului în crearea propriului mecanism în asimilarea unei materii independent și sigur;
- permite evaluarea independentă și individuală a elevilor;
- permite autoevaluarea;
- permite învățarea individuală independentă;
- permite adaptarea timpului studierii individuale a unei materii noi;
- permite evitarea subiectivismului în evaluare, etc.

Cu toate că importanța digitalizării nu mai stă sub semnul întrebării, rezultatele oferite în urma digitalizării lasă de dorit, în mare parte din cauza lipsei de resurse financiare sau din cauza numărului mic al cadrelor didactice ce posedă cunoștințe digitale competente în utilizarea conținuturilor digitale, ne mai vorbind de competența de creare a acestor conținuturi.

Pentru a soluționa măcar parțial acest aspect este nevoie ca anumiți actanți competenți în căutarea, prelucrarea și prezentarea informației, să pună la dispoziție cadrelor didactice anumite aplicații ce posedă o interfață cât se poate de simplă în utilizare, preferabil cu note informative despre utilizarea și manevrarea resurselor primite. Materialele fiind clare și să nu necesite lucru suplimentar de elaborare din partea cadrului didactic sau a elevului, fiind gata de utilizare pentru orice element activ al procesului educațional. Totodată aplicația trebuie să ofere materiale didactice în conformitate cu cerințele și standardele educaționale stabilite în acte de către ministerul de profil.

1. ANALIZA DOMENIULUI DE STUDIU

O mare parte dintre actorii implicați în procesul educațional caută metode de a integra noile tehnologiile în întregul proces al educației, dorindu-și o „clasă” cu soluții complete de digitalizare care să îi ajute cu ușurință pe profesori să transforme abilitățile profesionale în experiență educațională. Cadrele didactice, părinții și elevii au nevoie de soluții care să fie accesibile, ușor de folosit și de administrat și care să ofere un conținut educațional vast. Așadar, atât pentru profesori cât și pentru elevi noile tehnologii reprezintă o unealtă foarte importantă ce oferă numeroase beneficii, însă în anumite momente tehnologiile necesare presupun posedarea anumitor competențe digitale mai complexe.

Complexitatea competențelor digitale necesare redactării testelor variază în dependență de scopul utilizării și disciplina pe care o predă cadrul didactic. De exemplu, pentru un profesor de limbă și literatură română este mult mai simplu redactarea unui conținut științific de specialitate pentru un test, comparativ cu un profesor de matematică, fizică sau altă disciplină exactă ce necesită conținut științific specific disciplinei. Pentru a soluționa acest aspect este nevoie ca aplicațiile puse la dispoziția cadrelor didactice să posede o interfață cât se poate de simplă și clară în utilizare și cu note informative despre utilizarea și manevrarea resurselor primite, iar rezultatul sau produsul adică testul obținut să nu necesite modificarea, corectarea sau completarea acestuia fiind produs ce poate fi utilizat imediat.

Știind că „Evaluarea școlară este procesul prin care se delimitează, se obțin și se furnizează informații utile, permițând luarea unor decizii ulterioare.” [2], putem conchide că cadrul didactic trebuie să dispună de instrumente clare, ușor accesibile, manevrabile pentru realizarea unui test. Totodată evaluarea sau testul respectiv trebuie să ofere elevilor informații suficiente despre procesul de formare a competențelor specifice fizicii [1]. În funcție de aplicarea instrumentelor de evaluare de către profesor la disciplina Fizică, ca și la alte discipline de fapt, acesta va realiza următoarele tipuri de evaluare:

- evaluarea inițială – predictivă;
- evaluarea formativă – continuă;
- evaluarea sumativă – finală.

Deci evident este necesară posibilitatea obținerii testelor pentru orice tip de evaluare, cu orice conținut selectat anterior de cadrul didactic astfel încât testul să „facă o schimbare a paradigmei evaluării

– de la evaluare predominant sumativă la așa numitul model „evaluare pentru învățare, „evaluare pentru dezvoltare, „evaluare de formare”. [3] Pentru atingerea obiectivelor respective apare necesitatea prezenței în aplicație pentru selectare a diferitor tipuri de itemi. Totodată „elaborarea itemilor pentru conținuturile și obiectivele definite, trebuie proiectați pe trei niveluri de dificultate:

- nivel minim de dificultate;

- nivel mediu de dificultate;
- nivel superior de dificultate.

Itemii se aleg în așa fel încât să testeze o gamă largă de comportamente cognitive, de la cele de nivel inferior (cunoaștere, înțelegere) până la cele de nivel superior (analiză, sinteză). În acest scop se proiectează itemi: obiectivi, semiobiectivi, cu răspuns liber. [4] Deci testul trebuie să cuprindă itemi obiectivi, semiobiectivi, cu răspuns liber în dependență obiectivele propuse de cadrul didactic. Și deoarece scopul principal al evaluării este îmbunătățirea calității organizării și desfășurării procesului educațional, profesorul trebuie să aibă posibilitatea elaborării testului în așa fel încât să poată evalua cunoștințele și abilitățile practice ale elevilor la maximum, își formează o viziune clară asupra randamentului școlar, fapt ce îi permite atât lansarea unor programe compensatorii în activitatea cu elevii, cât și reproiectarea anumitor strategii didactice. Comparând rezultatele evaluării la diferite etape profesorul stabilește dinamica reușitei școlare, fapt pe care îl utilizează la elaborarea unor concluzii generale referitoare la eficiența activității sale didactice. Profesorul poate interveni corespunzător în formarea elevului și a corecta, acomoda sau în genere a modela procesul educațional planificat deoarece „Evaluarea este justificată și devine folositoare numai în măsura în care îl face pe profesor capabil să devină mai bun pentru ceea ce este chemat să facă” [5].

1.1 Analiza soluțiilor existente

„Paradigma docimologică a fost marcată în timp de următoarele patru concepții:

- evaluarea „comparativă”, a cărei funcție principală era de a compara elevii și de a-i clasifica, raportându-i pe unii la alții, acordându-le diplomă sau alte distincții după nivelul lor de reușită;
- evaluarea „prin obiective”, care are ca rol să furnizeze informații funcționale, permițând elevilor să se situeze în raport cu atingerea obiectivelor comune pentru toți elevii – (standarde unitare) și oferind soluții de ameliorare;
- evaluarea „corectivă”, care are ca scop să ofere elevului informații suplimentare în funcție de dificultățile constatate pentru a-i facilita învățarea;
- evaluarea „conștientizată” sau „formatoare”, care favorizează participarea activă și autonomia elevului, furnizându-i repere explicite, în scopul de a lua în mâini propria transformare, fiind conștient de propriile dificultăți și lacune” [6].

Concepțiile respective își găsesc locul în totalitate în sistemul educațional, acestea regăsindu-se în diferite modalități de testare a cunoștințelor ce corespund cerințelor curriculare. În ciuda acestui fapt soluțiile existente la moment, nu se bazează pe principiile și concepțiile recomandate pentru implementare cadrelor didactice, ci doar oferă un mecanism general ce poate fi folosit mai mult pentru întreținerea unei lecții sau pentru crearea unei lecții interactive, însă folosirea produselor existente nu face posibilă evaluarea egală, corectă din punct de vedere curricular.

Abordând problema evaluării, una din soluții ce ar asigura eficient cadrul didactic cu evaluări

scrise de orice tip într-un timp scurt și cu abilități digitale minime, crearea unei platforme educaționale specializate. La moment în Republica Moldova nu există o soluție pentru crearea unor teste de evaluare conform cerințelor curriculare și necesitățile speciale ale cadrului didactic, dar există unele alternative precum crearea quiz-urilor cu ajutorul diferitor aplicații web, una dintre acestea fiind ilustrat[în figura 1.1.1, sau redactarea testelor utilizând procesoare de text precum MS Office Word sau WPS Office, fiecare având pe lângă avantaje și anumite dezavantaje printre care putem menționa timpul suplimentar pentru prelucrarea, acomodarea conținutului, necesitatea posedării anumitor competențe speciale în domeniul TIC pentru prelucrarea conținutului, lipsa accesului la prelucrare etc. ceea ce îngreunează atingerea obiectivului propus de cadrele didactice sau uneori și de elevi.

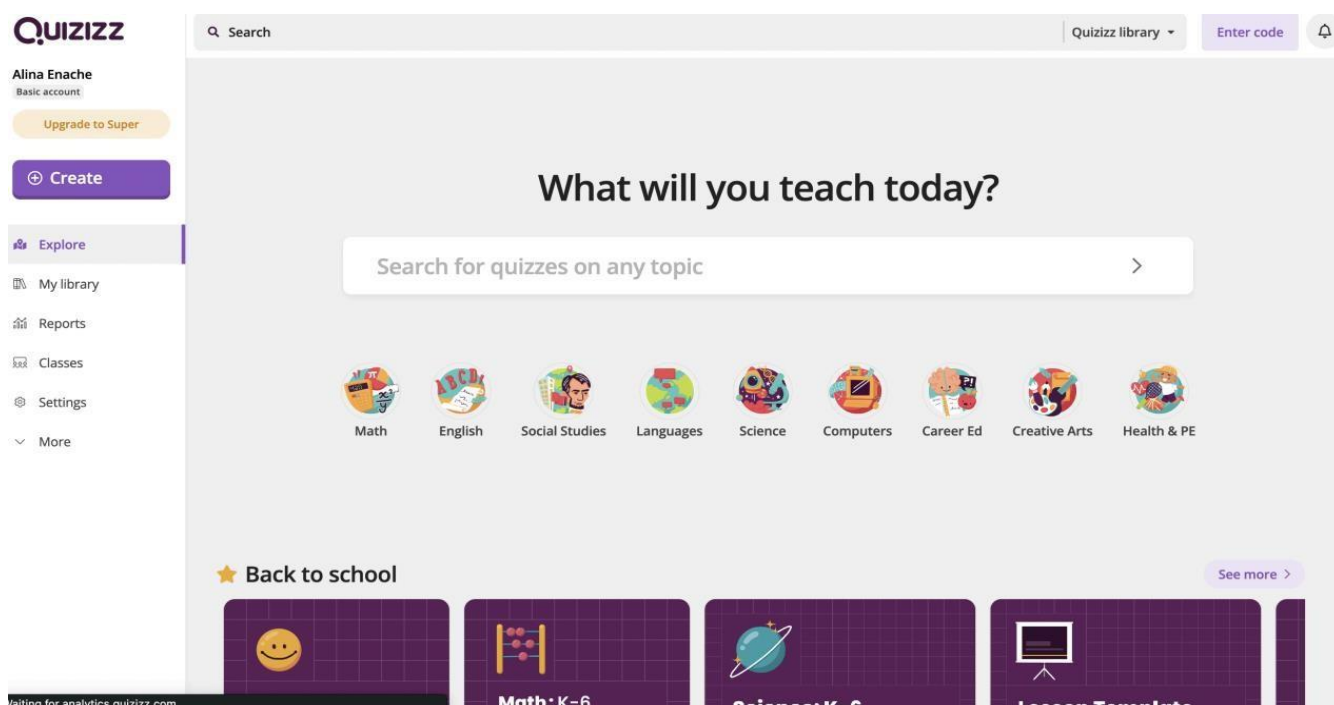


Figura 1.1.1 - Pagina principală quizizz.com

O soluție foarte convenabilă și atractivă pentru elevi este testarea cunoștințelor prin intermediul quiz-urilor, un exemplu de astfel de aplicație fiind site-ul quizizz, ilustrat în figura 1.1.1. Pentru o asemenea formă de evaluare a cunoștințelor există numeroase site-uri ce propun asemenea teste și simplifică elaborarea acestora, disponibile atât gratuit, cât și contra plată. Partea neplăcută în acest caz se reflectă asupra cadrului didactic, deoarece acest tip de teste în mare parte se limitează la exerciții de tipul selecție, pe când cadrul didactic are nevoie câteva categorii de itemi cu un diferit grad de complexitate în conformitate cu specificațiile curriculei disciplinei pe care o predă. De asemenea cadrul didactic trebuie să se asigure că fiecare elev în parte va avea la îndemână un dispozitiv ce îi va permite să efectueze testul și că toate aceste dispozitive să aibă posibilitatea de a se conecta la rețea. Deși în marea majoritate a cazurilor nu se întâmpină probleme la acest capitol, uneori pot apărea 6 probleme legate de calitatea conexiunii, ceea ce poate fi un impediment pentru realizarea testării, iar pe lângă aceasta, cadrul didactic nu este capabil să supravegheze activitatea elevilor pe toate dispozitivele, astfel

el nu poate fi sigur dacă aceștia au plagiat sau nu. Ca urmare a acestor factori, acest tip de teste cel mai adesea este folosit de către cadrele didactice pentru testele de sinteză la sfârșitul lecției sau ca o modalitate interactivă de verificare a temelor pentru acasă, însă nicidecum ca formă de evaluare inițială, periodică sau sumativă fie la finele unității de învățare sau finală.

Un alt neajuns al acestui produs se întâlnește în momentul în care cadrele didactice mai înaintate în vârstă apelează la acest mijloc de elaborarea a testelor. În cadrul acestei aplicații web utilizatorul dispune de o varietate mare în ceea ce privește tipul itemilor, după cum putem observa în imaginea 1.1.2.

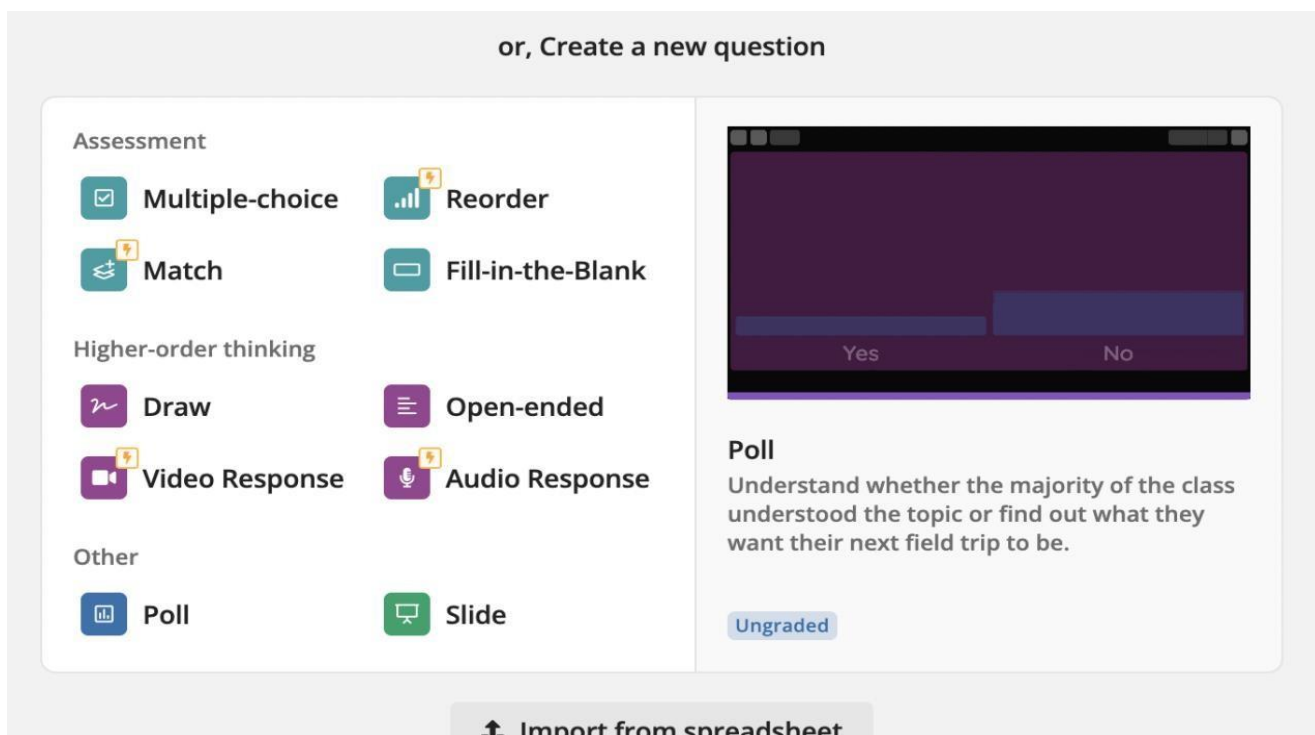


Figura 1.1.2 - Tipurile de itemi ce pot fi creați în cadrul testului

Cu toate acestea, crearea itemilor se dovedește a fi destul de dificilă pentru persoanele ce nu posedă cunoștințe digitale pe măsură. De asemenea eforturile depuse pentru crearea itemilor nu vor avea efect de lungă durată deoarece nu avem la dispoziție o opțiune pentru a genera mai multe variante în baza itemului creat, sau opțiunea de o stoca și refolosi în cadrul unui alt proiect.

De asemenea, un neajuns al acestui produs din perspectiva cadrelor didactice este faptul că cele mai interesante opțiuni ce oferă flexibilitate și asigură o oarecare diversitate adesea sunt contra plată, ceea ce pentru cadrul didactic este foarte neplăcut deoarece, în marea majoritate a cazurilor instituțiile de învățământ nu pot să compenseze aceste cheltuieli. Prin urmare acest neajuns demotivează digitalizarea procesului de evaluare și de predare-învățare.

Soluția la care apelează cel mai des cadrele didactice, este redactarea testelor cu ajutor unor procesoare de text, cel mai des MS Office Word, interfața căreia este ilustrată în figura 1.1.3. Pentru disciplinele cu profil umanist precum limba română, limba străină, istorie și geografie singurul

impediment ce poate fi întâlnit este dificultatea aranjării textului în pagină, însă în cazul disciplinelor cu profil real precum matematica, biologia, chimia și fizica lucrurile se complică. În cazul acestor discipline cadrul didactic are nevoie de cunoștințe digitale mai profunde întrucât este nevoit să opereze cu redactarea ecuațiilor, formulelor, limitelor, integralelor, etc. Pentru realizarea unui astfel de test cadrul didactic va avea nevoie de mai mult timp și mai multe eforturi în cazul disciplinelor reale comparativ cu cele umane, aceasta fiind și motivul pentru care marea majoritate a profesorilor preferă modalitatea tradițională de testare, în special cei mai învârstă.

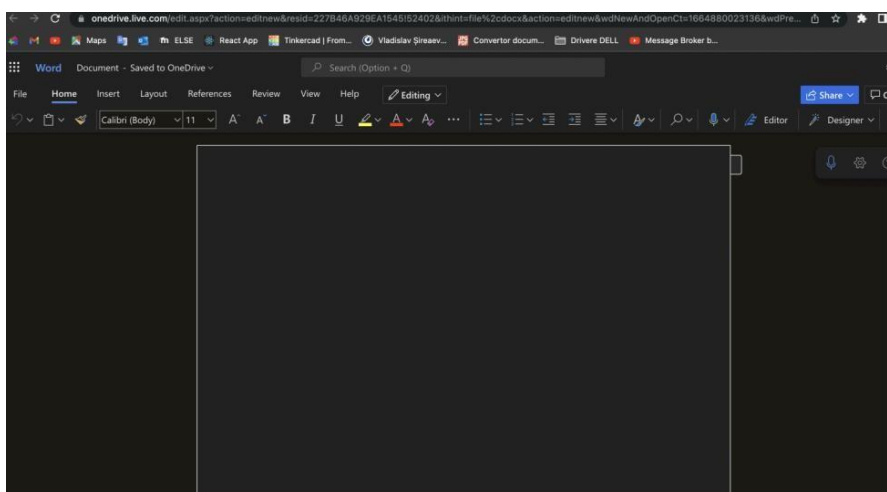


Figura 1.1.3 - Procesorul de text online MC Office Word Live

Acest produs, deși este, de asemenea, un produs web, are neajunsuri pentru persoanele ce se încadrează în grupul cadrelor didactice cu capacități digitale slabe. Pentru disciplinele umaniste crearea testelor de evaluare fie de tip inițială, fie formativă sau sumativă, este destul de simplu, deoarece cadrele didactice trebuie să culeagă la tastatură conținutul testului și dacă este cazul să îl aranjeze în pagină prin folosirea listelor numerotate sau nenumerate sau în rare cazuri folosirea tabelelor, care de asemenea nu este dificilă, opțiunile enumerate mai sus fiind accesibile din meniul Acasă sau în cazul tabelelor în meniul Inserare. În cazul disciplinelor cu profil real precum Fizica, Matematica și Chimia situație este îngreunată de ecuațiile, expresiile matematice și diferite simboluri ce fac parte din vocabularul specific limbajului. Pentru crearea acestora este nevoie de a opera cu opțiunile Ecuații și Simboluri accesibile din meniul Inserare. Interfața acestor opțiuni fiind redată în figura numărul 1.1.4.

Meniurile acestor opțiuni sunt uneori destul de greu de aranjat în pagină și cadrele didactice întâmpină dificultăți în utilizarea acestora deoarece sunt numeroase manevre ce trebuie memorate pentru elaborarea unui sistem de ecuații și/sau inecuații, pentru utilizarea limitelor și integralelor sau pentru crearea graficelor (care aici trebuie creată prin desene) în cazul matematicii, pentru ecuațiile și simbolurile în cazul Fizicii, etc.

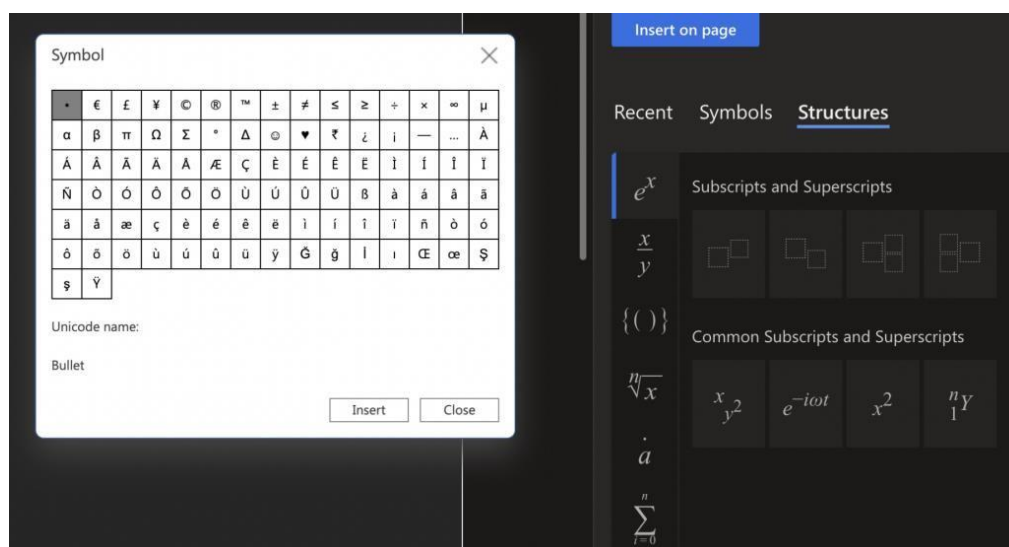


Figura 1.1.4 – Interfața opțiunilor Simbol și Ecuatii

Metodele enumerate mai sus, deși sunt soluții parțiale ale problemei, acestea nu ușurează deloc activitatea cadrului didactic, care este nevoit să depună eforturi semnificative pentru crearea unui test ce va corespunde cu cerințele curriculare în totalitate. Începând cu selectarea tipurilor de itemi potriviți și până la selectarea conținutului itemilor respectivi, cadrul didactic trebuie să cerceteze literatura de specialitate și să își perfecționeze competențele digitale pentru a putea lucra cu un anumit program ce îi permite realizarea testului dorit.

Soluțiile enumerate mai sus au un scop destul de restrâns ce se limitează la un anumit aspect ce diferă de la o soluție la alta. Astfel profesorii sunt nevoiți să opereze cu mai multe soluții, deoarece la moment nu există una ce să combine toate necesitățile cadrului didactic în ceea ce privește evaluarea cunoștințelor elevilor.

Deși sunt centrate pe un domeniu mai restrâns, soluțiile existente au o mulțime de neajunsuri din cauza cărora cadrul didactic nu poate fi sigur că toți elevii sunt verificați în situații egale. De exemplu, cea mai răspândită soluție la care cadrele didactice apelează fie din obișnuință, fie din lipsa unei alternative, o reprezintă culegerile de teste sumative și/sau formative. Cel mai mare dezavantaj al acestei soluții este faptul că cadrul didactic nu poate să limiteze accesul elevilor la testele de evaluare, astfel unii pot să îl rezolve acasă de câteva ori, sau pe maculator, etc. De asemenea, există și dezavantaje comune atât pentru elevi cât și pentru studenți, și anume faptul că această culegere de teste este în formă fizică, ceea ce înseamnă că fiecare elev va fi nevoit să ducă în rucsac câte o culegere de teste pentru fiecare disciplină la care este preconizată testarea, pe lângă celelalte manuale și materiale necesare pentru lecțiile respective. Pentru cadrele didactice, de asemenea, este dificil deoarece acesta este nevoit să ducă cu el câte 25-30 de culegeri cu teste pentru fiecare clasă pe lângă celelalte materiale ale sale. Aceste incomodități și neajunsuri complică activitatea cadrului didactic, atât pe plan financiar, dat fiind faptul că aceste culegeri trebuie procurate de către părinții elevilor ce nu mereu sunt de acord cu acest lucru, iar unii pur și simplu nu au posibilitatea de a procura câteva culegeri pentru fiecare

disciplină, cât și pe plan estetic, întrucât uneori cadrul didactic ajunge să fie confundat cu un hamal, din cauza a toane de caiete ce verificate, toate de culegeri, planșe și alte materiale necesare pentru întreținerea lecției și încă și culegerile de teste.

Nu doar soluția analizată mai sus are neajunsuri, ci și toate celelalte soluții existente, deoarece la moment nu se focusează și nu se investește pe cât este necesar în modernizarea sistemului educațional și în digitalizarea conținutului educativ și informativ.

Produsele recomandate de minister sunt destul de învechite, ceea ce cauzează numeroase neplăceri cadrelor didactice dat fiind faptul că conținutul și modulele materiei au suferit schimbări. Prin urmare cadrele didactice sunt nevoie să adapteze materialele oferite la necesitățile lor.

Soluția respectivă este o unealtă ce are ca scop principal ușurarea procesului de creare a testului care va permite realizarea a orice tip de evaluare conform cerințelor curriculare și a ghidului de implementare a curriculei fără a fi necesară utilizarea altor produse soft.

Acest lucru va fi mult mai ușor pentru cadrele didactice deoarece toate întrebările pentru unitățile de conținut vor fi accesibile prin intermediul unei singure aplicații, ce va permite realizarea oricărui test doar prin metoda selecției. Prin urmare simplitatea utilizării va fi încă un aspect pozitiv al acestui produs ce îi va oferi un avantaj față de produsele deja existente.

Un alt avantaj este faptul că autorul itemilor din această aplicație, decide independent cine poate avea acces la itemii creați sau la testele create, ceea ce pe viitor va permite ca testele să fie verificate și notate în mod automat prin intermediul baremului de notare și baremului de verificare.

1.2 Scopul și obiectivele propuse

Scopul principal al acestui proiect este elaborarea unei aplicații web pentru generarea automată a testelor de evaluare pentru disciplinele cu profil real, ce ar permite:

- obținerea testelor necesare cu minim de abilități digitale;
- obținerea testului într-un timp restrâns;
- posibilitatea selectării itemilor de către cadrele didactice;
- aranjarea conținutului sub forma machetei;
- introducerea itemilor în motorul de generare digitală de testelor;
- informarea mediului educațional și încurajarea utilizării aplicației de către profesori pentru valorificarea cunoștințelor și abilităților la disciplina fizica, prin rezolvarea situațiilor semnificative propuse în aplicația dată.

Proiectul respectiv este o aplicație web single-page, creată cu ajutorul unui modul cu suport oficial, ce va oferi un setup modern pentru Build, ceea ce va ușura mentenanța și permite focusarea pe implementarea funcționalului produsului.

Utilizarea acestui produs va ajuta cadrul didactic în pregătirea pentru lucrările de evaluare fiind un suport ce permite atât crearea materialelor necesare în baza selecției cadrului didactic dar și

automatizează generarea documentelor necesare pentru verificarea și notarea lucrărilor de evaluare atât inițiale cât și formativă și sumativă. Acest lucru este unul extrem de important deoarece astfel cadrulul didactic îi rămâne mai mult timp pe care îl poate petrece fie pregătind materiale adăugătoare pentru lecțiile ulterioare, fie ajutând elevii ce au nevoie de consultații.

În timp platforma respectivă poate fi modificată pentru a aduce mai multe beneficii în domeniul educației precum :

- accesibilitate: Studenții pot accesa testele online din orice locație și la orice oră, prin intermediul dispozitivelor lor personale. Nu mai este necesar să se deplaseze la o locație fizică pentru a lua testul;

- economisirea timpului: Testele online pot fi finalizate mult mai rapid decât cele tradiționale, deoarece studenții nu mai trebuie să completeze manual răspunsurile în hârtie și să aștepte să fie evaluate. Rezultatele testului pot fi disponibile în timp real sau într-un interval de timp scurt după ce testul a fost finalizat;

- imposibilitatea copiatului: Testele online sunt mai dificil de copiat, deoarece acestea pot fi concepute pentru a împiedica studenții să copieze prin intermediul software-ului de detectare a plagiatului sau prin utilizarea de întrebări randomizate;

- flexibilitatea întrebărilor: Testele online pot fi configurate pentru a prezenta întrebări diferite pentru fiecare student, astfel încât să fie dificil să se copieze întrebările. De asemenea, testele online pot fi configurate pentru a permite feedback-ul imediat asupra răspunsurilor sau pentru a oferi explicații suplimentare asupra răspunsurilor corecte.

1.3 Importanța și posibilitățile sistemului

Întrucât „orice activitate umană conștientă conține o etapă de verificare a rezultatelor, în vederea comparării cu scopul acesteia, pentru ameliorarea acțiunii în etapa următoare. Procesul de învățământ nu se poate nici el lipsi de verificare și apreciere, evaluarea rezultatelor școlare constituind o componentă importantă a activității didactice, utilă atât pentru cel care conduce nemijlocit procesul, cât și pentru beneficiarii acestuia – elevii.” [7. Pag.145] Evaluarea corectă și univocă a elevilor este importantă deoarece, reușita academică este criteriul de bază al selecției ce are loc la etapa de admitere în instituțiile superioare, prin urmare trebuie să ne asigurăm că toți elevii au fost supuși unei evaluare ce corespunde cerințelor curriculare în mod egal, având același grad de dificultate, aceleași timp disponibil pentru realizare și aceleași resurse disponibile pentru efectuarea testului. Acest lucru ar garanta obiectivitatea mediilor de intrare în diferite instituții de învățământ, media pentru solicitarea burselor de merit, etc.

Evaluarea reprezintă, alături de predare și învățare, o componentă operațională fundamentală a procesului de învățământ [6]. În sistemul educațional evaluarea, ne informează despre eficiența strategiilor și metodelor de predare-învățare, dar în același timp aceasta reflectă și corectitudinea obiectivelor operaționale, în dependență de rezultatele evaluării, prin urmare este necesar ca fiecare test

elaborat de cadrul didactic să îndeplinească cerințele stipulate în ghidurile de implementare a curriculei și alte acte normative ce specifică modalitatea sau conținutul ce trebuie să constituie obiectul evaluării.

Evaluarea trebuie concepută nu numai ca un control al cunoștințelor sau ca mijloc de măsurare obiectivă, ci ca o cale de perfecționare, ce presupune o strategie globală a formării [6]. Reieșind din aceasta, toate testele de evaluare la care este supus elevul trebuie să fie bine concepute și să urmărească un scop și obiectiv bine definit pentru a putea fi parte din formarea continuă a elevului, deoarece pentru a vedea eficiența programei curriculare și a ghidurilor de implementare, cadrul didactic trebuie să își desfășoare activitatea în strictă concordanță cu prevederile documentelor și actelor normative, ceea ce înseamnă că testele elaborate nu trebuie să se abată de la cerințele curriculare.

Problematica evaluării în procesul de învățământ solicită cadrului didactic atitudini și demersuri de înțelegere semnificativ mai profunde și complexe. Este nevoie de atitudini proactive și de prelucrare științifică reflectivă a informațiilor didactice psihopedagogice, precum și de dezvoltare a competenței de evaluare a viitorului cadru didactic [6]. Pentru o mai bună funcționare a sistemului de învățământ din Republica Moldova, activitatea cadrului didactic trebuie să fie centrată pe analiza elevilor iar pregătirea pentru lecțiile sale trebuie să nu se centreze pe învățarea mereu a noilor tehnologii necesare pentru a atinge parțial scopul dorit. Pentru cadrele didactice este necesară o soluție unică ce va permite minimizarea eforturilor pentru crearea varietății în ceea ce ține de numărul itemilor și conținutul acestora, adică cadrului didactic să i se ofere posibilitatea de a crea variante multiple sau diferite tipuri de teste fără ca acesta să fie nevoit să învețe un alt mecanism de creare a testelor sau alte tehnologii noi ce îi vor permite să elaboreze același lucru. Crearea unei astfel de soluții ar permite micșorarea timpului necesar cadrului didactic pentru crearea unui text ce va corespunde cu toate cerințele curriculare și ale ghidurilor de implementare.

Un alt aspect îl reprezintă timpul necesar pentru crearea documentației anexă, precum baremul de notare, baremul de corectare, etc. Deși munca cadrelor didactice se bazează pe interacțiune cu elevii, pe lângă orele propriu zise, cadrele didactice au îndatorirea de a îndeplini numeroase acte normative și documente precum planuire de lună durată, planuire de scurtă durată, dări de seamă, etc.

Strategiile de evaluare reprezintă manierele operaționale de stabilire a formelor și tipurilor de evaluare, metodelor, tehnicilor și probelor de evaluare a randamentului școlar, a modalităților de îmbinare a acestora și momentelor în care se aplică, descriptorilor de performanță, baremelor și sistemelor de notare [8]. Metoda de evaluare este o cale prin care cadrul didactic oferă elevilor posibilitatea de a demonstra nivelul de stăpânire a cunoștințelor, de formare a diferitelor capacități, testate prin utilizarea unei diversități de instrumente adecvate scopului urmărit. Prin stabilirea și aplicarea unei sau altei metode, evaluatorul declanșează și orientează obținerea unor informații (răspunsuri) în corespundere cu obiectivele de evaluare [9]. La moment modalitățile de evaluare disponibile alcătuiesc un ansamblu metodologic coerent, ce are caracter sistemic, ceea ce ne permite să

le facem obiectul automatizării, astfel acest lucru ar permite crearea deprinderilor de autoanaliză în rândul elevilor/studentilor și va simplifica activitatea cadrelor didactice prin generarea baremului de notare și corectare.

Pentru profesori această aplicație face posibilă generarea diferitor variante cu ușurință, precum și crearea testelor fulger pentru verificarea cunoștințelor obținute în urma lecției respective. De asemenea profesorul obține un număr foarte mare de itemi ce pot fi dați elevilor pentru exersare. Forma prestabilită a documentului permite generarea baremului de notare și baremului de corectare cu foarte mare ușurință, economisind timpul și resursele cadrului didactic, ce pot fi direcționate în alte direcții, precum explicarea temelor sau interacțiunea cu elevii.

Pentru elevi acest mecanism de generare a testelor aduce câteva beneficii foarte importante, precum claritatea condițiilor și sarcinilor ce necesită de a fi realizate, forma asemănătoare cu examenele de stat, ce permite acomodarea cu sistemul de corectare și notare. De asemenea aceștia obținut numeroase variante pentru exersare.

Importanța acestei aplicații constă în faptul că la elaborarea unui test este nevoie în primul rând de timp și desigur anumite competențe fie digitale fie specifice disciplinei fizic, ceea ce este foarte dificil pentru cadrele didactice ce predau disciplina dată după recalificare venind din alt domeniu sau predarea fizicii ca disciplină secundară ne fiind specialist. Această aplicație permite obținerea diferitor tipuri de teste în scurt timp indiferent de competențele digitale. Această aplicație își îndeplinește scopul articolului de a ajuta toate fără excepție cadrele didactice în obținerea unui număr mari de evaluări pentru un număr mare de elevi, elaborarea testelor cu grad diferit de complexitate, elaborarea testelor de evaluare inițială, formativă și sumativă.

Organizarea aplicației este una clasică, corectă din punct de vedere științific, clară și accesibilă la utilizare. Astfel, cadrul didactic are acces atât la elaborarea testului cât și la tabelele necesare pentru soluționarea situațiilor semnificative din această aplicație și nu numai. Cadrul didactic are acces la elaborarea testului pe două căi accesând butoanele situate în poziția de sus a paginii sau a ecusonului specific testului. Condițiile de parcurgere a mersului elaborării testului sunt clare, corecte și ușor realizabile atât pentru cadre didactice cât și pentru elevi. Testul poate fi imprimat cât și salvat în dispozitivul individual al cadrului didactic. Tema analizată este importantă prin prisma faptului că permite economia timpului cadrului didactic la elaborarea testului. Este o aplicație utilă pentru toate 12 cadrele didactice fie cu experiență, tinere, după recalificare, care citesc disciplina fizica ca ne fiind specialist, etc. De asemenea aplicația oferă o soluție adecvată, reală și gratuită pentru cadrul didactic privitor la o problemă veche, dar care rămâne de actualitate, testul de evaluare electronic.

Aspectele pozitive ale acestei aplicații sunt:

- economisirea timpului profesorului pentru crearea unui test;
- oferirea unui aspect prietenos pentru elevi;

–testele pot fi generate de către profesori fără a fi nevoiți să cunoască aplicații pentru editarea textelor precum MS. Word, Pages, etc;

–automatizarea baremelor de notare și corectare ce vor permite profesorului cu ușurință să verifice testele iar studenților să se autoevalueze;

–forma pe care se va baza va fi asemănătoare cu cea de la examenele de bacalaureat, ceea ce va permite acomodarea studenților cu aceasta;

–după același mecanism vor putea fi generate 3 tipuri de teste în dependență de necesitățile profesorilor.

Crearea unei platforme pentru generarea testelor de evaluare sumativă și periodică va permite elaborarea conținutului conform cerințelor, și anume testul va conține 24% teme de cunoaștere și înțelegere, 44% itemi de aplicare și 325 itemi de integrare.

După logare, cadrele didactice vor avea posibilitatea de a introduce exemplele proprii de itemi ce vor fi vizibili pentru alți profesori, pentru alți elevi sau doar pentru autor, alegerea fiind la discreția utilizatorului. Acest lucru va permite generarea variantelor multiple pentru testele de evaluare cu eforturi minimale și crearea a diferitor teste pentru exersare.

Utilizatorul va avea posibilitatea să salveze machetele create pentru a le accesa ulterior, în cazul în care va avea necesitatea de a repeta testul sau de a face o lecție de analiză. De asemenea, dacă utilizator va fi conectat ca cadru didactic, ca anexă pentru testul generat va fi adăugat baremul de notare și baremul de corectare, ceea ce va reduce din timpul cadrelor didactice necesar pentru elaborarea testelor, oferindu-le posibilitatea de a genera teste după o formă prestabilită, utilizând atât conținutul introdus de către utilizator, cât și conținutul introdus de alți profesori ce au oferit acces.

După accesarea site-ului va fi posibilă crearea conturilor în calitate de elev sau în calitate de profesor. În calitate de elev, utilizatorul va fi capabil să genereze teste din itemii publici pentru a exersa, sau de a efectua online testul creat de profesor, pe când în calitate de profesor, utilizatorul va fi capabil să introducă itemi noi privați (vizibili doar pentru el), combinați (vizibil atât pentru elevi cât și pentru profesori) și publici (vizibil doar pentru profesori).

Forma de bază a testului propus ca produs final al acestei aplicații va fi sub aspectul unui tabel cu 3 coloane, prima fiind completată cu numărul itemilor, iar a treia cu punctajul acordat pentru fiecare item sub forma L 0 1 2 3 4

În cea de a doua coloană vor fi plasați itemii evaluarea sumativă respectivă, sub formă de casete cu două componente de bază: prima va fi condiția ce va fi statică și se va popula după categoria itemului din baza de date, iar a doua va conține conținutul itemului propriu zis. Condiția itemului va fi evidențiată prin scrisul mai pronunțat și va urma puțin spațiu liber după ea. Conținutul itemilor va fi ilustrat cu un scris simplu și va fi interpretat după nevoile de reprezentare a fiecărei categorii de itemi.

Pentru itemii cu adevărat și fals, caseta cu conținut va fi împărțită în două coloane în proporție de

95% și 5%, prima fiind completată cu afirmația a cărei valoare de adevăr trebuie analizată, iar a doua cu simbolurile A pentru adevărat și F pentru fals.

În cazul itemilor de tip completarea caseta va fi o celulă unică în care va fi ilustrat conținut itemilor după ce va avea loc substituirea simbolul specific pentru desemnarea spațiilor de completare cu un anumit spațiu necesar pentru fiecare cuvânt. Itemii de tip corespondență vor avea celula de bază divizată iarăși în două coloane în proporții egale. Prima coloană va fi populată cu mărimile fizice a căror unitate de măsură

trebuie identificată, iar a doua cu unitățile de măsură. Alinierea conținutului se va face în asemenea mod, pentru a permite efectuarea corespondenței între elementele din prima coloană cu elementele din a doua coloană prin intermediul săgeților sau liniilor.

Pentru problemele cu grad scăzut de complexitate, celula de bază va fi populată cu două rânduri, prima cu condiția problemei, iar a doua cu numărul de linii necesare pentru rezolvarea acestui item. Acest număr va fi citit din baza de date.

În cazul problemelor cu un grad mai înalt de dificultate celula de bază va fi populată cu trei rânduri. Primul pentru condiția problemei, al doilea pentru imagine anexă a problemei respective, ce va fi afișată în dependență de dimensiunile stipulate la crearea itemului, iar cel de-al treilea va fi spațiul de rezolvare a itemului.

1.4 Cerințe funcționale și nefuncționale

Elaborarea platformei de gestionare a testelor de evaluare sumativă și periodică pentru disciplinele școlare cu profil real a avut la bază următoarele cerințe nefuncționale:

–cerințe de conformitate:

Produsul respectiv este protejat prin lege conform următoarelor prevederi:

- a) codul contravențional al Republicii Moldova nr. 218-XVI din 24.10.2008, art. 96, 400;
- b) codul penal al Republicii Moldova nr. 985-XV din 18.04.2002, art. 1851 , 1853.

Acest produs este subiectul următoarelor legi:

- a) hotărârea Guvernului nr. 774 din 13.08.1997 cu privire la Nomenclatorul serviciilor cu semnificație juridică în domeniul protecției obiectelor proprietății intelectuale, cu modificările ulterioare;
- b) hotărârea Guvernului nr. 89 din 10.02.2012 pentru aprobarea Regulamentului cu privire la înregistrarea obiectelor dreptului de autor și drepturilor conexe, Monitorul Oficial nr. 34-37/114 din 17.02.2012.

–cerințe arhitecturale:

Produsul respectiv este o aplicație web ce poate fi accesată de pe orice dispozitiv, inclusiv telefonul mobil, folosind un browser din următoarea listă:

- a) Firefox, versiunea 105.0.3 și mai sus;

- b) Samsung Internet, versiunea 17.0.7 și mai sus;
- c) Chrome, versiunea 105 și mai sus;
- d) Opera, versiunea 85 și mai sus;
- e) Safari, versiunea 15.6 și mai sus;
- f) Edge, versiunea 100 și mai sus;
- g) Brave.

Accesarea produsului respectiv prin intermediul unui alt browser ce nu este inclus în lista de mai sus, poate duce la un comportament nedorit și neașteptat a produsului, de aceea este necesar ca browser- ul folosit să fie actualizat până la ultima versiune și să fie oferită permisiunea pentru utilizarea javascript- ului.

Serverul acestei aplicații rulează pe sistemul de operare GNU/Linux, pe când aplicația de pe partea de client poate fi utilizată de orice dispozitiv indiferent de sistemul de operare a acestuia.

–cerințe de interfață externă:

- a) interfețe cu utilizatorul:

Pentru a satisface nevoile utilizatorilor produsul dispune de o vastă gamă funcțională menită să simplifice utilizarea produsului respectiv. Astfel utilizatorul are la dispoziție funcțional ce permite generarea a diferite tipuri de teste, precum și generarea documentației afiliate lucrării conform cerințelor curriculare, iar toate funcționalități respective vor fi accesibile prin intermediul apăsării unor butoane sugestive.

Utilizatorul va fi obligat să se autentifice sau să se înregistreze pentru a utiliza funcționalitățile de bază a produsului. După această etapă utilizatorul va putea accesa toate resursele produsului în dependență de rolul atribuit contului său.

Indiferent de rolul atribuit contului, toți utilizatorii au acces la funcționalitățile de bază ce țin de gestionarea testelor de evaluare, precum crearea testelor, salvarea și printarea acestora.

Pentru elaborarea unui teste de evaluare nou, utilizatorul va trebui să aleagă clasa și unitatea de conținut pentru care se realizează evaluarea. După care utilizatorul va alege din fiecare categorie de itemi, ce vor fi arătați secvențial, selectând itemii pe care dorește să îi includă în testul de evaluare.

În baza selecției utilizatorului, va fi generat un test de evaluare ce va fi alcătuit din toți itemii selectați, amplasați după macheta utilizată pentru elaborarea examenelor de stat și bacalaureatului. De asemenea va fi posibilă salvarea testului pentru a fi folosit ulterior, precum și selectarea formatului dorit pentru salvarea documentului pe dispozitiv propriu, ceea ce va permite utilizatorului să efectueze anumite modificări manual, în cazul în care acesta întâlnește această necesitate/ dorință.

Pentru printarea documentului utilizatorul va trebui să dispună de un dispozitiv de printare configurat și compatibil cu dispozitivul electronic pe care a accesat aplicația web sau pe care are salvată varianta finală a testului;

b) interfețe hardware:

Produsul respectiv este o aplicație web ce poate fi accesată de pe orice dispozitiv, inclusiv telefonul mobil, folosind un browser ce trebuie să fie actualizat până la ultima versiune și să fie oferită permisiunea pentru utilizarea javascript-ului.

Serverul acestei aplicații rulează pe sistemul de operare GNU/Linux, pe când aplicația de pe partea de client poate fi utilizată de orice dispozitiv indiferent de sistemul de operare a acestuia;

c) interfețe software:

Pentru primirea anumitor notificări despre actualizări și funcționalitățile noi este necesar ca utilizatorul manual să permită accesul instrumentului de navigare folosit.

De asemenea pentru buna funcționare a produsului este necesar ca utilizatorul să permită utilizarea javascript-lui, pentru a asigura afișarea corectă a componentelor și aranjarea corectă a componentelor produsului;

d) protocoale și interfețe de comunicare:

Produsul respectiv este o aplicație web ce folosește HTTPS, care este un protocol securizat ce încapsulează protocolul HTTP. Acesta va asigura siguranța datelor prin intermediul unui flux SSL/TLS care criptează datele ce provin de la un browser web la un server web.

Partea client este realizată folosind Angular, este un cadru de proiectare a aplicațiilor și o platformă de dezvoltare pentru crearea de aplicații eficiente și sofisticate pe o singură pagină. Backend-ul în schimb va fi realizat folosind C# web API, unde API este un set de definiții de sub-programe, protocoale și unelte pentru programarea de aplicații și software. Un API poate fi pentru un sistem web, sistem de operare, sistem de baze de date, hardware sau diferite biblioteci. De exemplu, când este vorba despre interfața dintre programele de aplicație și sistemul de operare, acesta stabilește în amănunt modul în care programele de aplicație pot accesa (apela) serviciile sistemului de operare sub care rulează.

—alte cerințe nefuncționale:

a) cerințe de performanță:

Pentru buna funcționare a acestui produs, utilizatorul trebuie să fie conectat obligatoriu la o rețea de internet cu conexiunea puternică. De asemenea este important ca ecranul dispozitivului utilizat de către utilizator să folosească un dispozitiv performant pentru a beneficia din plin de toate posibilitățile platformei respective.

Pentru ilustrarea adecvată a adătelor și componentelor aplicației web este necesar ca instrumentele de navigare web să permită utilizarea javascript-ului și să fie mereu actualizate până la ultima versiune stabilă.

Dat fiind faptul că aplicația respectivă este o aplicație web, conexiunea la internet este elementul cheie, de care depinde calitatea produsului. Ca urmare a acestui fapt, pentru a asigura integritatea

datelor și finisarea cu succes a procesului, este necesar ca pe parcursul utilizării funcționalului de generare a testului, utilizatorul să asigure o conexiune bună și continuă, în caz contrar varianta finală a testului poate să nu includă toți itemii selectați de utilizator sau să fie redactat greșit din punct de vedere structural, generând erori;

b) cerințe de siguranță:

Produsul respectiv nu folosește datele personale în alte scopuri decât cele specificate la etapa de înregistrare, și anume comunicarea identității între cadrul didactic și elevii săi, prin urmare utilizatorii trebuie să introducă date veridice pentru a nu întâmpina neplăceri sau probleme pe parcursul utilizării aplicației respective. Odată ce datele au fost validate ele vor putea fi modificate doar de către administratorul platformei.

Pentru achitarea abonărilor sunt necesare ca datele din contul utilizatorului să fie veridice, iar cardul utilizat pentru achitare trebuie să nu fie expirat sau invalid pentru a evita blocarea contului de utilizator, ce ulterior poate fi deblocat doar de către administrator;

c) cerințe de securitate:

Pentru a evita apariția erorilor în procesul de generare a testelor de evaluare este necesar ca utilizatorul să asigure o bună conexiune la internet. De asemenea este necesar ca contul utilizatorului să fie creat folosind date veridice, întrucât acestea sunt verificate în procesul de generare a testului, dar și în alte momente precum adăugarea conținutului, crearea modelului de test, etc.

Modificarea ulterioară a datelor contului nu poate fi efectuată decât prin intermediul administratorului;

d) atribute de calitate:

Interfața grafică a utilizatorului este una minimalistă, ce permite utilizarea simplă a funcționalului. Marea majoritate a componentelor vizuale din cadrul produsului au fost dezvoltate independent de către echipa tehnică, pentru a corespunde tuturor cerințelor și pentru a permite crearea unei interfețe cât mai simplă.

Funcționalul produsului respectiv permite gestionare deplină a procesului de generare a testelor de evaluare, începând cu selectarea conținutului testului, și până la selectarea tipului dorit la salvarea fișierului nou creat;

e) cerințe față de documentare:

Documentația produsului respectiv este una simplistă ce ajută la înțelegerea modului de funcționare precum și a modului de utilizare a produsului respectiv, termenii tehnici fiind explicați în detaliu. De asemenea documentația include un model de test elaborat folosind aplicația respectivă, explicat din punct de vedere structural și funcțional.

Pe lângă acestea, documentația include și un ghid de utilizare foarte detaliat, ce explică fiecare parte a funcționalului disponibil. Acesta cuprinde și imagini ce au menirea de a îl ghida pe utilizator în

cazul în care acesta întâmpină dificultăți în utilizarea interfeței propuse.

De asemenea pe lângă aceste cerințe nefuncționale, aplicația web elaborată a încapsulat și următoarele cerințe funcționale:

–este necesar ca utilizatorul să își creeze un cont fie în calitate de cadru didactic, fie în calitate de elev, având asociată o poștă electronică unică. În funcție de tipul contului, utilizatorul va putea accesa diferite rute, iar cele comune nu vor avea toate funcționalitățile disponibile în cazul în care utilizatorul va avea rolul de elev;

–pentru a corespunde cu cerințele curriculare este nevoie ca selecția utilizatorului să se încadreze în unul dintre tipurile de evaluare standarde: inițială, formativă sau sumativă. Reieșind din aceasta selecția utilizatorului trebuie să fie dirijată pentru a comunica utilizatorului la ce etapă se află și ce tipuri de evaluări are posibilitatea să creeze utilizând itemii selectați, iar pentru asta utilizatorul va primi anumite indicii ce îi vor ușura utilizarea produsului respectiv;

–produsul respectiv nu va accepta diferite tipuri de teste ce nu corespund cerințelor, prin urmare utilizatorii nu vor fi capabili să creeze teste de evaluări cu un conținut ce nu se încadrează conform curriculei în niciunul dintre tipurile de evaluare ce pot fi aplicate pentru verificarea cunoștințelor.

1.5 Caietul de sarcini

Procesul de elaborare a sistemului de generare a testelor de evaluare cuprinde trei faze principale, și anume: faza introductivă, faza de elaborare și faza de încheiere / de finisare a platformei de gestionare a testelor de evaluare sumativă și periodică pentru disciplinele cu profil real.

Faza introductivă este alcătuită din analiza domeniului de studiu a proiectului, ce cuprinde descrierea cerințelor și elaborarea template-urilor pentru testele de evaluare, ce ulterior vor fi folosite pentru structurarea documentului.

În faza de realizare a sistemului, are loc procesul de dezvoltarea a codului sistemului din care va rezulta aplicația finală ce va permite generarea mai multor tipuri de teste. La aceasta etapă mai are loc și elaborarea și efectuarea testelor pentru validarea funcționalului descris în etapa de modelare și crearea bazei de date, pentru stocarea eficientă și sigură a conținutului și datelor utilizatorilor.

În faza de finisare are loc elaborarea și crearea instrucțiunilor necesare pentru utilizarea corectă a sistemului creat și pentru acomodarea cu funcționalitățile acestuia și modul de funcționare. Toate aceste lucruri se documentează și se alcătuiește ulterior manualul utilizatorului, unde se explică clar cum trebuie să fie utilizat sistemul și ce posibilități oferă acesta și documentația tehnică ce cuprinde mai multe detalii despre arhitectura și componența sistemului.

Reieșind din acestea caietul de sarcini are structura specificată în tabelul 1.5.1:

| Nr. crt. | Tematica lucrărilor preconizate | Termene planificate | |
|---------------------|--|-----------------------------|----------------------------|
| | | început/ sfârșit | numărul de zile |

| | | | |
|---|--|--|---------|
| 1 | <p><i>Analiza domeniului de studii și Modelarea și proiectarea sistemului informatic.</i></p> <ul style="list-style-type: none"> - descrierea domeniului și tipul aplicației precum și sfera de activitate de care ține acest proiect; - argumentarea importanței platformei realizate și stabilirea principalelor avantaje oferite de acest produs comparativ cu alte platforme/ produse cu obiective similare; - stabilirea cerințelor atât funcționale cât și nefuncționale; - specificarea funcționalităților de bază și principiului de funcționare al produsului realizat. | <p>30.01.2023</p> <p>-</p> <p>10.02.2023</p> | 10 zile |
| 2 | <p><i>Realizarea sistemului</i></p> <ul style="list-style-type: none"> - crearea modelelor pentru stocarea itemilor; - crearea bazei de date; - conectarea aplicației la baza de date; - definirea regulilor de accesare. | <p>20.02.2023</p> <p>-</p> <p>03.03.2023</p> | 10 zile |
| 3 | <p><i>Realizarea sistemului</i></p> <ul style="list-style-type: none"> - crearea paginilor; - definirea rutelor; - crearea componentelor; - crearea serviciilor; - afișarea conținutului primit din baza de date; - implementarea generării documentului în funcție de selecția utilizatorului; - vizualizarea documentului generat; - salvarea fișierului generat. | <p>06.03.2023</p> <p>-</p> <p>14.04.2023</p> | 30 zile |
| 4 | <p><i>Realizarea raportului și prezentării</i></p> <ul style="list-style-type: none"> - testarea platformei/sistemului; - fixarea erorilor; - crearea documentației; - finalizarea raportului și prezentării. | <p>17.04.2023</p> <p>-</p> <p>19.05.2023</p> | 25 zile |

Tabelul 1.5.1- Caietul de sarcini

2. MODELAREA ȘI PROIECTAREA SISTEMULUI INFORMATIC

Funcționarea aplicației depinde mult de datele de intrare primite de la utilizatori, ceea ce complică proiectarea, dar și înțelegerea deplină a principiilor ce stau la baza platformei. Întrucât utilizatorii dispun de mai multe posibilități în ceea ce privește modul de utilizare, este foarte important ca mecanismele interne ale aplicației să fie bine gândite astfel încât să nu cadă în momentul în care nu primește informația deplină de la utilizator sau primește informația incorectă, dar și limitarea informației ce poate fi utilizată ca date de intrare trebuie să fie limitată.

Un alt aspect este planificarea structurii aplicației, deoarece această depinde de foarte mulți factori ce posedă caracteristici diferite. Conținutul platformei trebuie să fie elaborat conform cerințelor specifice domeniului în parte, dar în același timp trebuie și să nu încalce cadrul legal ce face referire la tipul respectiv de informație.

Astfel pentru crearea unei aplicații de succes este necesare o bună planificare și detaliere a acesteia, înainte de a începe implementarea și dezvoltarea acesteia. Ca urmare a acestui fapt se va efectua o analiză complexă ce va permite elaborarea diagramei de context, ce va evidenția detaliile și rolurile acestora în procesle platformei de gestiune.

Pentru modelarea platformei de gestionare a testelor de evaluare sumativă și periodică va fi utilizat limbaj UML - Unified Modeling Language, care este un limbaj standard pentru descrierea de modele și specificații pentru software, acesta fiind creat de către consorțiul Object Management Group (OMG) care a mai produs printre altele și standardul de schimb de mesaje între sisteme CORBA. UML a fost la bază dezvoltat pentru reprezentarea complexității programelor orientate pe obiect, al căror fundament este structurarea programelor pe clase, și instanțele acestora (numite și obiecte). Cu toate acestea, datorită eficienței și clarității în reprezentarea unor elemente abstracte, UML este utilizat dincolo de domeniul IT. Așa se face că există aplicații ale UML-ului pentru management de proiecte, pentru business Process Design etc.

Diagramele vor fi create folosind aplicația web `app.diagrams.net` ce posedă o interfață clară și simplă în utilizare.

Subiectul modelării este sistemul ce stă la baza platformei de gestionare a testelor de evaluare, alcătuit din principalele funcționalități ale platformei. Această parte a sistemului este cea mai cerută în rândul cadrelor didactice, ceea ce o face principalul element ce garantează succesul aplicației, prin urmare necesită o planificare detaliată și bine gândită.

Scopul modelării sistemului este ilustrarea tuturor componentelor ce participă la procesul de generare a testelor de evaluare în funcție de selecția utilizatorului. De asemenea, modelarea va permite cititorilor să înțeleagă principiul de funcționare al proceselor ce stau la baza platformei. Modelarea componentelor va reflecta interacțiunea elementelor componente ale sistemului precum și a celor exterioare acestuia din punctul de vedere al administratorului.

Pentru realizarea fluxului aplicației, cererile utilizatorilor și necesitățile cadrelor didactice vor fi intrările, acestea fiind informația ce urmează să fie procesată de aplicație, iar în urma parcurgerii fluxului aplicația va oferi utilizatorilor, testul generat după forma și formatul selectat de utilizator, precum și anexele necesare după caz.

Ca reglementări, vor fi utilizate curriculum națională și ghidul de implementare, iar ca mecanisme vor fi utilizate baza de date și alte software-uri necesare pentru stocarea și securizarea datelor.

Principalele activități ce se realizează la parcurgerea fluxului platformei de generare a testelor depind de 4 elemente și anume: cadrele didactice, administratorul, arhitecții și sistemul de generare a fișierelor. Acestea interacționează cu aplicația în diferite momente ale fluxului de utilizare sau de creare a platformei.

Activitățile indispensabile pentru funcționarea platformei și realizarea scopului acesteia sunt:

- generarea testului în baza selecției utilizatorului de către sistem;
- adăugarea conținutului în baza de date de către cadrele didactice;
- cadrele didactice selectează itemii, pe care îi doresc în componența viitorului test;
- administratorul validează identitatea cadrelor didactice;
- sistemul generează anexele pentru testele de evaluare anexate;
- sistemul stochează variantele de teste realizate anterior;
- sistemul limitează accesul utilizatorilor la resursele platformei cu nivel de accesibilitate limitat;
- administratorul răspunde cererilor/solicitărilor;
- utilizatorii formulează cereri.

2.1 Descrierea comportamentală

Pentru înțelegerea principiului de funcționare a platformei de gestionare a testelor de evaluare sumativă și periodică pentru disciplinele școlare cu profil real este necesară modelarea detaliată a tuturor comportamentelor posibile a platformei. Prin urmare cu ajutorul limbajului UML se descrie:

– imaginea generală asupra sistemului (Use Case Diagram)

Pentru a reprezenta posibilitățile utilizatorilor de a interacționa cu sistemul respectiv, adică cu platforma de gestionare a testelor de evaluare pentru disciplinele școlare cu profil real, și anume autentificarea în sistem, vizualizarea itemilor, crearea itemilor noi, și salvarea testelor generate de sistem, a fost creată diagrama generală de interacțiune a utilizatorului cu produsul realizat, ilustrată în figura 2.1.1. În diagramă am reprezentat relațiile de asociere între actorul Users și use-case-urile Login, ItemsList, TestContent și TestsList.

Toate aceste posibilități sunt disponibile utilizatorilor platformei, indiferent de rolul pe care îl are utilizatorul, dar diferă interfața și meniul de acțiuni disponibile pentru a manevra resursa asupra căreia are acces utilizatorul respectiv.

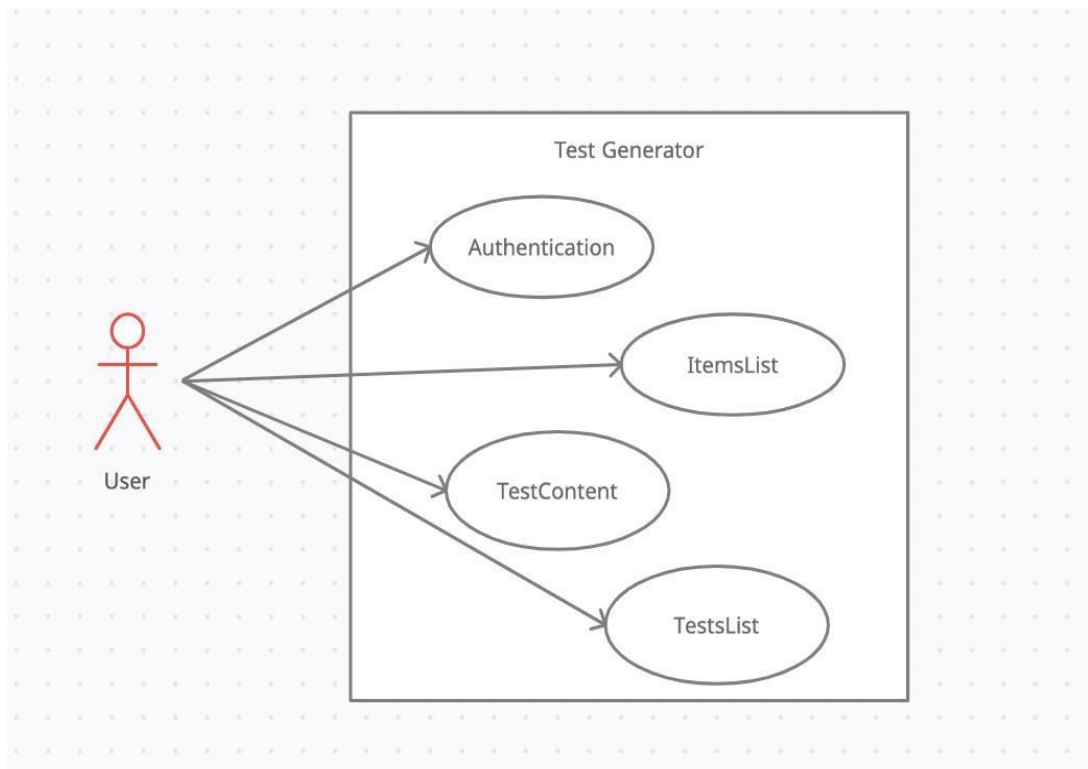


Figura 2.1.1 - Diagrama generală de interacțiune a utilizatorului cu sistemul

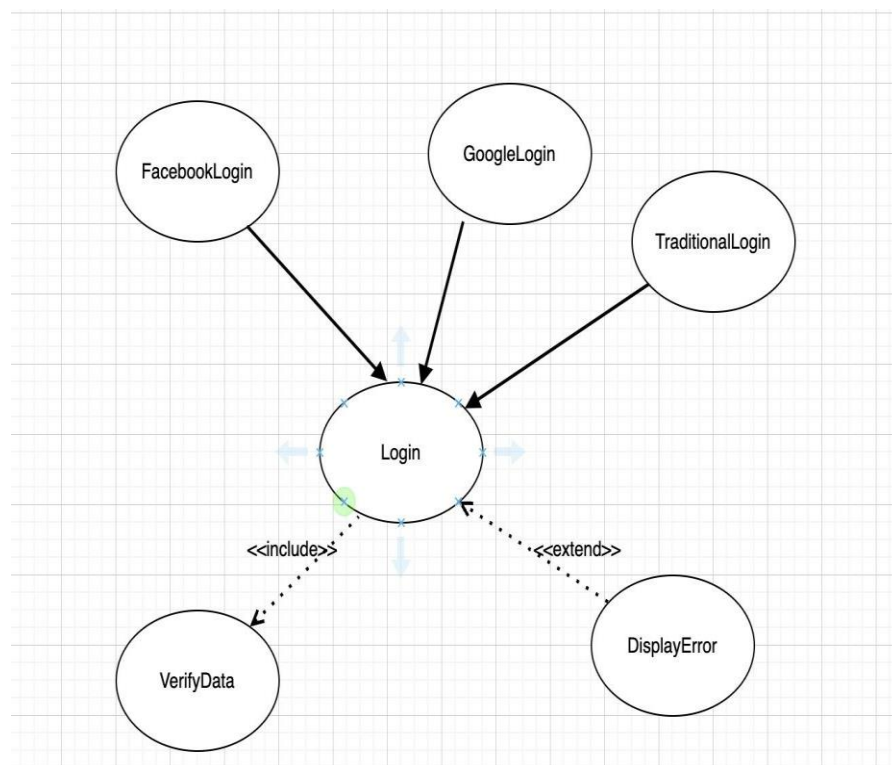


Figura 2.1.2 - Diagrama cazului de utilizare Login (autentificarea)

O altă diagramă este ilustrată în figura 2.1.2, ce reprezintă structural și schematic procedura de autentificare parcursă de clienții/utilizatorii sistemului. Pentru aceasta utilizatorului îi este oferită posibilitate de a crea cont folosind datele din Facebook, Google sau tradițional - prin crearea unui nume de utilizator și a unei parole, ceea ce am reprezentat prin relația de moștenire în care cazurilor de utilizare

FacebookLogin, GoogleLogin și TraditionalLogin moștenesc de la cazul de utilizare Login.

După aceasta datele introduse vor fi obligatoriu procesate pentru a verifica validitatea acestora (include VerifyData), iar dacă vor fi depistate careva erori acestea vor fi afișate (extend DisplayError).

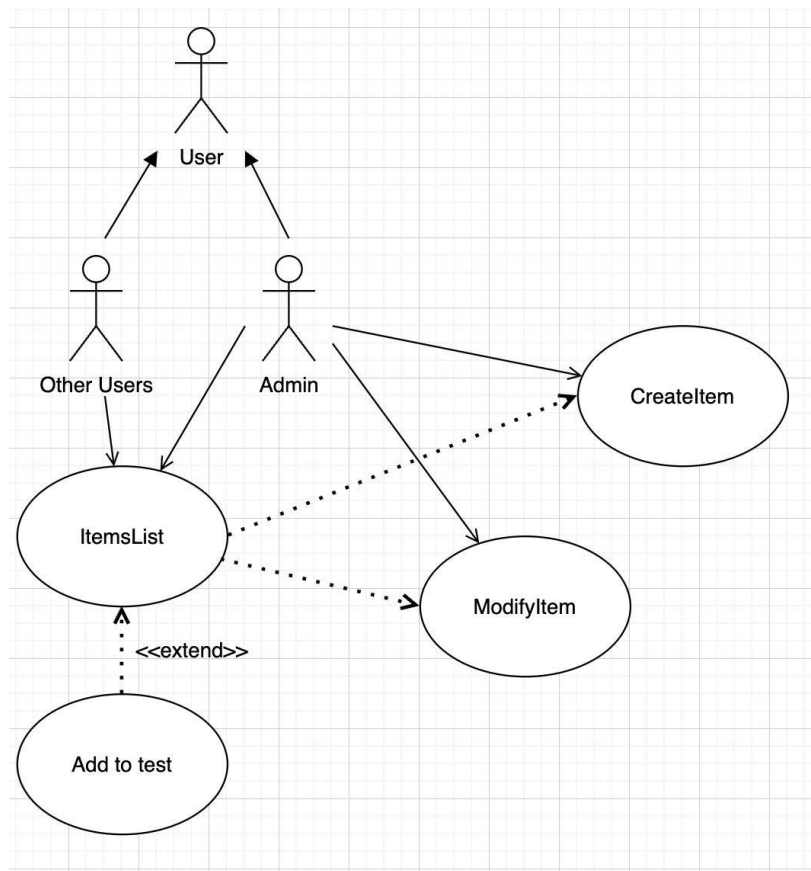


Figura 2.1.3 - Diagrama cazului de utilizare ItemsList (Lista cu itemi/conținuturi)

Conform diagramei din figura 2.1.3, posibilitatea de a modifica itemii o vor avea doar acei utilizatori ce au cont de administrator, pe când accesul la lista de itemi o au toți utilizatorii. De asemenea modificarea și crearea unui item va duce la modificarea listei cu produse deci cazul de utilizare ItemList depinde de cazul de utilizare ModifyItem și CreateItem.

Pe lângă acestea la dorința utilizatorului, el va putea sa adauge unul sau mai mulți itemi în conținutul testului adică în lista cu itemi ce va sta la baza generării testului.

Pentru ilustrarea acestor relații am făcut ca cazul de utilizare Admin și otherUsers să moștenească de la User prin relația de generalizare; Adminul să interacționeze prin relația de asociere cu ModifyItem/CreateItem și ItemList, iar ceilalți utilizatori de asemenea prin relația de asociere doar la ItemList, în timp ce între use-case-ul ItemList și ModifyItem este relația de dependență. Ca extensii (relația de dependență de tip extend) use-case-ul ItemList va avea posibilitățile de adăugare în conținutul testului.

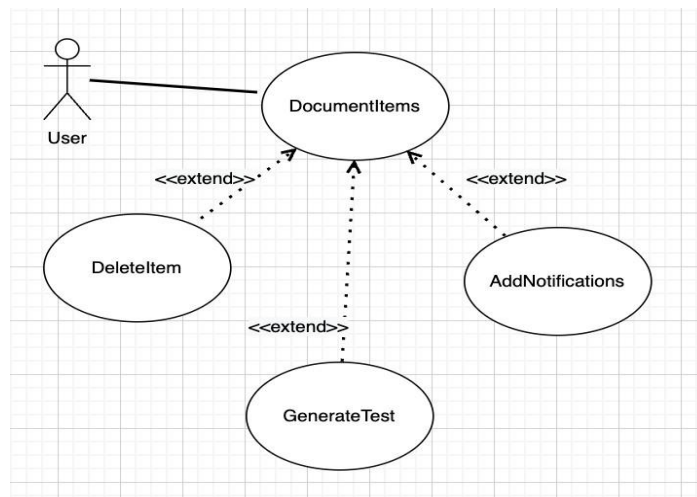


Figura 2.1.4 - Diagrama cazului de utilizare lista cu itemi

În figura 2.1.4 avem reprezentate posibilitățile utilizatorilor de a șterge unul sau mai mulți itemi din listă, generarea în format PDF sau Word și adăugarea unei notificări pentru a urmări descărcările sau eventualele modificări ce pot apărea în urma editării itemului de către administrator ceea ce am ilustrat prin relația de dependență extend între cazul de utilizare DocumentItems și cazurile DeleteItem, Generate Test și AddNotifications.

–modelarea vizuală a fluxurilor (Activity diagram)

Din figura 2.1.5, observăm că pentru generarea unui test de evaluare este necesar ca mai întâi să fie selectați itemii sau itemul dorit în listă cu conținut pentru document. După aceasta utilizatorul poate verifica conținutului listei și confirma sau modifica ordinea itemilor în listă prin apăsarea unui buton. În urma acestui pas se va stabili ordinea fiecărui item în viitorul document, după care utilizatorul va trebui să selecteze formatul dorit pentru viitorul document. La finalizarea acestei activități motorul sistemului va genera documentatul ce va conține itemii selectați de utilizator și va fi posibil de salvat și/sau descărcat în formatul indicat de către utilizator.

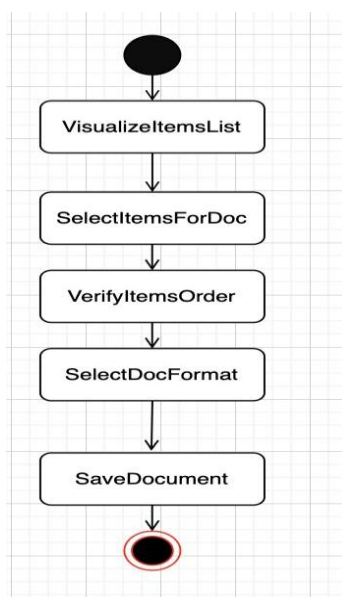


Figura 2.1.5 - Activitățile procesului de generare

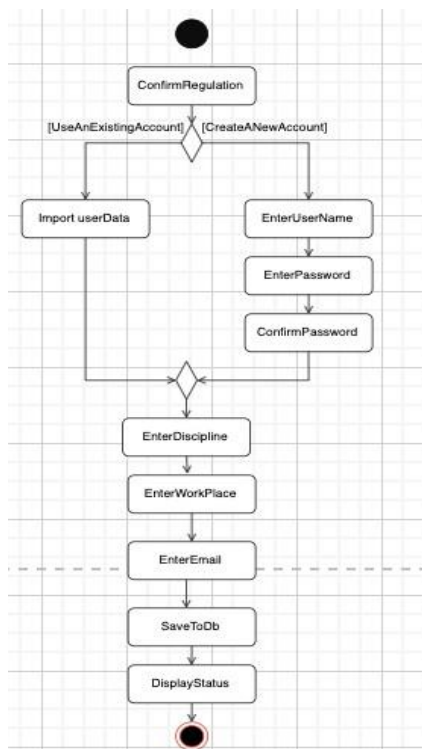


Figura 2.1.6 - Activitățile din cadrul înregistrării

Conform diagramei din figura 2.1.6, pentru crearea unui cont utilizatorul va putea alege din mai multe opțiuni, printre care utilizarea unui cont deja existent de la care se va prelua numele de utilizator și parola sau metoda tradițională prin care utilizatorul va crea și introduce manual numele de utilizator și parola pe care ulterior o va confirma. După activitatea respectivă, utilizatorul trebuie să indice locul de muncă, precum și disciplina pe care o predă, urmată de email de la muncă. Ca urmare se va crea o cerere de adăugare a utilizatorului în baza de date, iar ulterior se va afișa statul cererii utilizatorului pentru a oferi mai multă comoditate utilizatorului și pentru a îl informa despre statutul contului și eventualele neclarități în cazul în care acestea vor apărea.

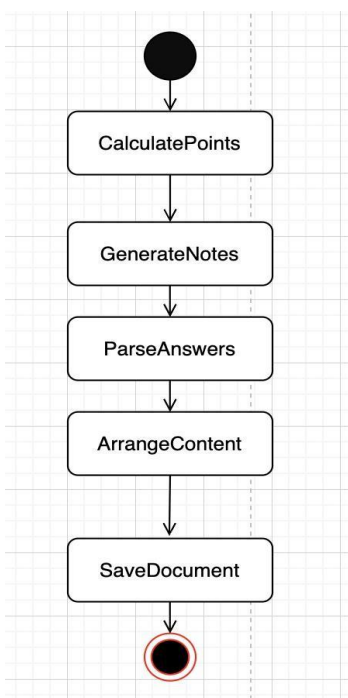


Figura 2.1.7 - Procesul de generare a baremelor

Reieșind din diagrama din figura 2.1.7, pentru a genera baremul de notare și baremul de corectare este necesar ca utilizatorul să apese pe butonul adăugare anexe înainte de salvarea documentului cu testul de evaluare. După apăsare se va calcula punctajul total al testul de evaluare și în funcție de procente stabilite de administrator se vor genera intervalele de puncte pentru fiecare notă, ceea ce va reprezenta baremul de notare.

La finalizarea acestei activități se va efectua parsarea pentru stabilirea corelației dintre fiecare item al testului de evaluare și subpunctele ce trebuiau îndeplinite pentru a găsi soluția problemei.

La următoarea etapă va avea loc aranjarea itemilor și subpunctelor asociate acestora după valoarea indexului fiecăruia. La final vor fi generate două anexe: baremul de notare și baremul de corectare, pe care utilizatorul le va putea utiliza ca template sau documente auxiliare testului de evaluare, iar în cazul în care utilizatorul va întâmpina necesitatea de a adăuga mai multe detalieri la conținutul baremelor, acesta va putea să le descarce în format odt sau word pentru a le modifica după caz.

–stările de tranzacție a sistemului (Statechart diagram)

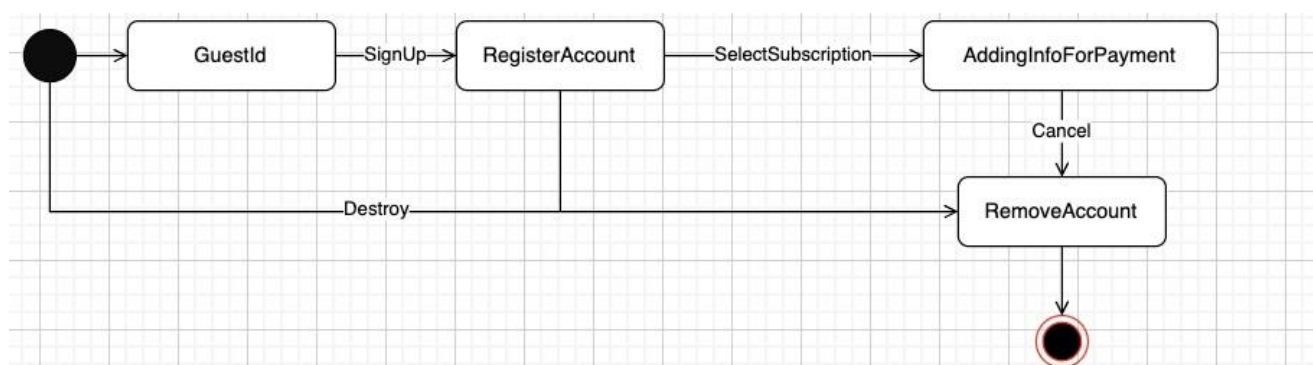


Figura 2.1.8 - Diagrama de stare a procesului de creare a contului

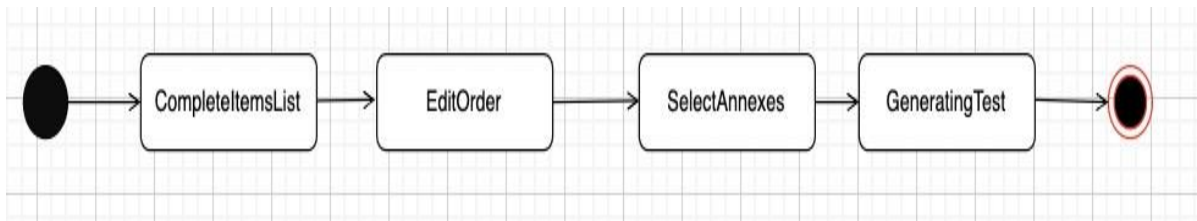
În diagrama din figura 2.1.8 este ilustrat procesul de creare a unui cont care începe prin crearea unui Id temporar pentru vizitatori (GuestId). În această stare utilizatorul se poate afla câteva zile, până când timpul de inactivitate depășește timpul permis pentru sesiune, caz în care automat se șterge din baza de date acest Id pentru a putea fi reutilizat ulterior, sau când dorește să se înregistreze prin crearea unui cont stabil (RegisterAccount state).

La intrarea în această stare este nevoie de selectarea unei metode de autentificare, după care inputul utilizatorului este parsat și validat iar la ieșire se adaugă în baza de date.

După finalizarea procesului de înregistrare și autentificare cu succes utilizatorul poate accesa resursele sistemului în dependență de rolul atribuit acestuia de către administrator sau poate selecta un abonament lunar pentru a mări numărul itemilor și alte funcționalități ale sistemului.

Pentru aceasta va fi necesar să introducă metoda de plată (AddInfoForPayment State) și informațiile necesare pentru efectuarea plăților și anume numele băncii, tipul cardului, numărul cardului, codul de verificare, acceptarea validărilor etc. După aceasta prin selectarea subscripției și a termenului

dorit poate să utilizeze funcționalitățile sistemului. Aici de asemenea dacă nu va fi înregistrată o activitate pe o perioadă îndelungată (mai mare decât 18 zile) contul va fi șters pentru a evita suprasolicitare bazei de date (RemoveAccount State).



Din figura 2.1.9, observăm că pentru generarea testului de evaluare utilizatorul trebuie să selecteze anumite tipuri de itemi și un număr prestabilit de itemi de fiecare tip, pentru completarea listei de itemi (CompleteItemsList State) după care va urma procedura de editare și aranjare ce se va solda cu obținerea unei confirmări privind validitatea datelor selectate și compatibilitatea acestora cu tipul de evalaure dorit.

După aceasta va urma starea de așteptare în care se vor selecta anexele necesare și se va verifica posibilitatea generării acestora cu lista de conținut existentă la momentul respectiv, după care conținutul se trimite generatorului pentru creare fișierului ce va putea fi salvat sau descărcat în funcție de preferințele utilizatorului și de dispozitivul utilizat de acesta.

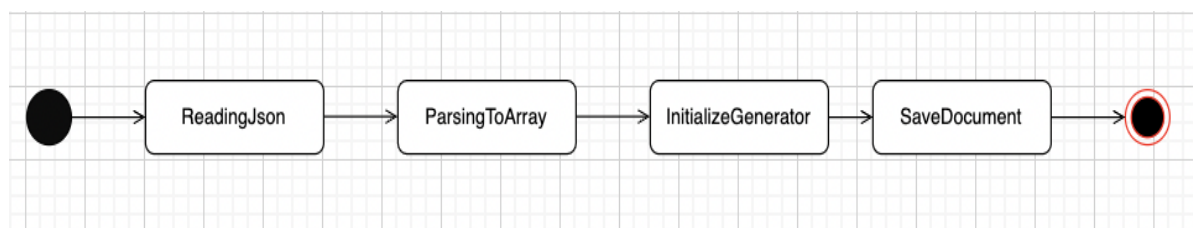


Figura 2.1.10 - Diagrama de stare a procesului de generare a testului

Generarea testului va avea la bază mai multe elemente indispensabile, reprezentate în diagrama de stări din figura 2.1.10. Procesul de generare va începe prin citirea informației primite de la utilizator prin intermediul obiectului de tip Json, ce va conține atât anexele dorite, cât și conținutul propriu-zis al testului de evaluare. După citirea informației va avea loc parsarea acesteia pentru a crea un model compatibil cu cel ce stă la baza modulului utilizat pentru generarea testului. După finalizarea cu succes a parsării, va avea loc inițializarea modulului generatorului cu obiectul rezultat în urma parsării pentru a obține testul de evaluare, dar și anexele dorite. La finisarea acestei stări va fi creat un document cu testul de evaluare și încă un document ce va conține anexele selectate de utilizator. După aceasta ele vor fi salvate în baza de date pentru a putea fi descărcate de utilizator.

–descrierea scenariilor de utilizare a aplicațiilor (Sequence diagram)

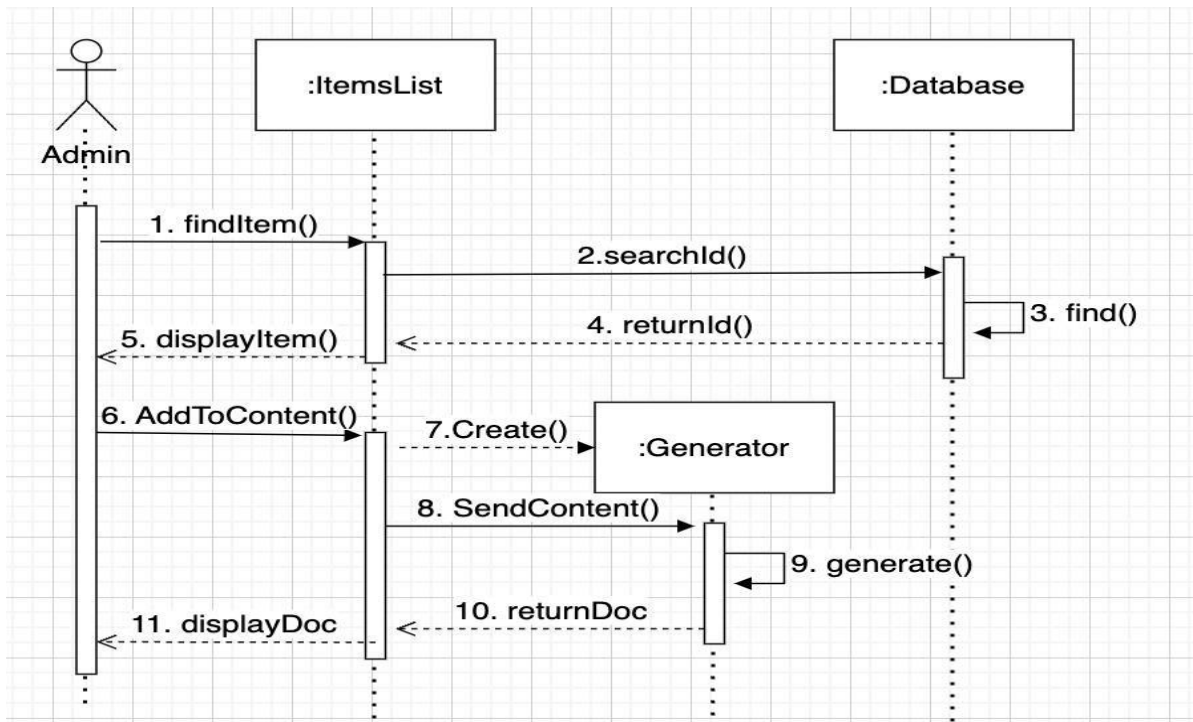


Figura 2.1.11 - Adăugarea unui produs în coșul de cumpărături

În diagrama de secvență 2.1.11 am ilustrat procesul de adăugare al unui item în conținutul testului de evaluare. Inițial userul trimite un mesaj de tip call către lista de itemi, pentru a găsi itemii doriți, după care obiectul listă de itemi trimite un mesaj de tip call către baza de date pentru a găsi Id-ul itemului.

Odată ce focus-controlul a ajuns la obiectul baza de date acesta inițiază un autoapel (recursion) pentru găsirea Id-ului care este transmis către lista de obiecte ca răspuns la apelul inițiat anterior. După ce lista de itemi primește Id-ul el afișează informația despre item, ca răspuns la solicitarea user-ului

Având la dispoziție datele despre itemi utilizatorul decide să adauge itemii în con cu cumpărături, prin urmare inițiază obiectul BookList este nevoit să inițializeze crearea unui nou obiect (create() :Cart). Odată ce obiectul CoșDeCupărături a fost creat, obiectul listăDeCumpărături trimite către el Id-ul printr-un mesaj de tip call, care este salvat prin intermediul unui autoapel (recursion).

După aceasta printr-un mesaj de tip call solicită, către obiectul baze de date, salvarea modificărilor. Când obiectul baze de date devine activ acesta inițializează un proces intern de actualizare a datelor (recursion update()), iar după finalizarea acestuia returnează un mesaj de succes către obiectul :Cart (return Success()), după care obiectul coș de cumpărături returnează către obiectul listă de cărți, noua listă a produselor ce fac parte din coș (return SendNewList()). După aceasta obiectul ca răspuns la solicitarea utilizatorului, este afișat conținutul coșului de cumpărături (return DisplayCartList()).

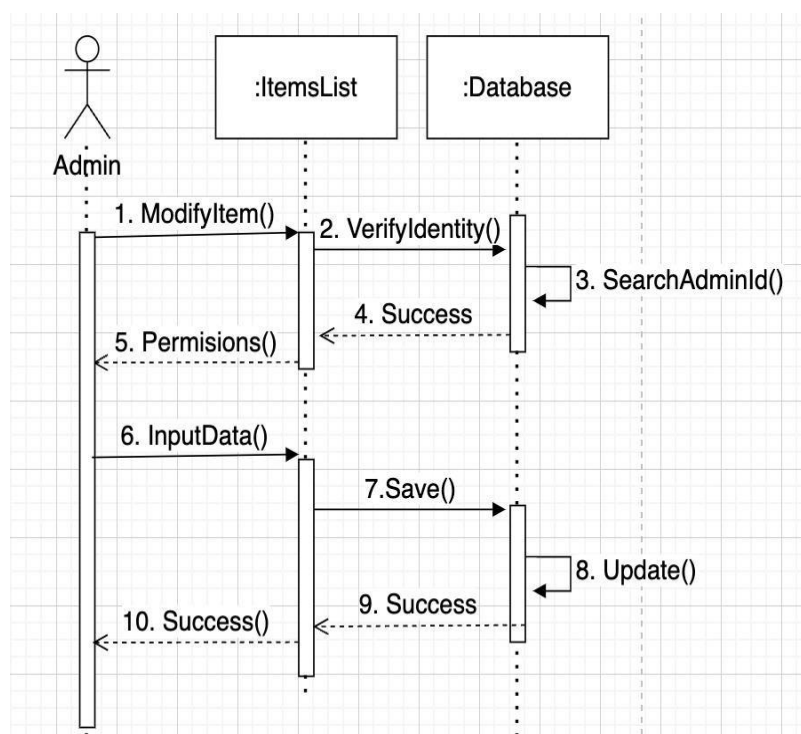


Figura 2.1.12 - Modificarea datelor despre o carte

În diagrama 2.1.12 am reprezentat procesul de modificare a datelor despre un produs din lista cărților, în care participă 3 entități: utilizatorul, lista de produse și baza de date.

Pentru aceasta, odată ce obiectul admin ce moștenește din clasa users inițiază un apel către entitatea :BookList, apare necesitatea verificării identității utilizatorului. Ca urmare, :BookList trimite un apel către baza de date pentru verificarea identității utilizatorului, aceasta fiind realizat printr-un autoapel ce are ca scop căutarea id-ului utilizatorului în lista administratorilor (recursion SearchAdminId()). După aceasta baza de date returnează un mesaj de succes către obiectul clasei listă de produse, care la rândul său confirmă permisiunea de a modifica datele drept răspuns către utilizator (return Permission()).

După aceasta administratorul trimite, iarăși prin intermediul unui apel către entitatea lista cu cărți (call InputData()), datele ce dorește să le modifice. În momentul în care entitatea :BookList primește acest mesaj, ea trimite modificările către baza de date așteptând un răspuns de la aceasta (call SendChanges()). La rândul ei baza de date își actualizează datele prin intermediul unui autoapel (recursion Update()) și trimite un mesaj de succes către lista cu cărți (return Success() către :BookList). După aceasta :BookList trimite un mesaj de finalizare a procesului cu succes prin intermediul unui retur (return Success() către Admin).

–fluxurile de mesaje și legăturile dintre componentele sistemului (Collaboration diagram)

Procesul de înregistrare, reprezentat în figura 2.1.13 care poate fi realizat pe două căi: prin intermediul unui administrator sau independent de către client.

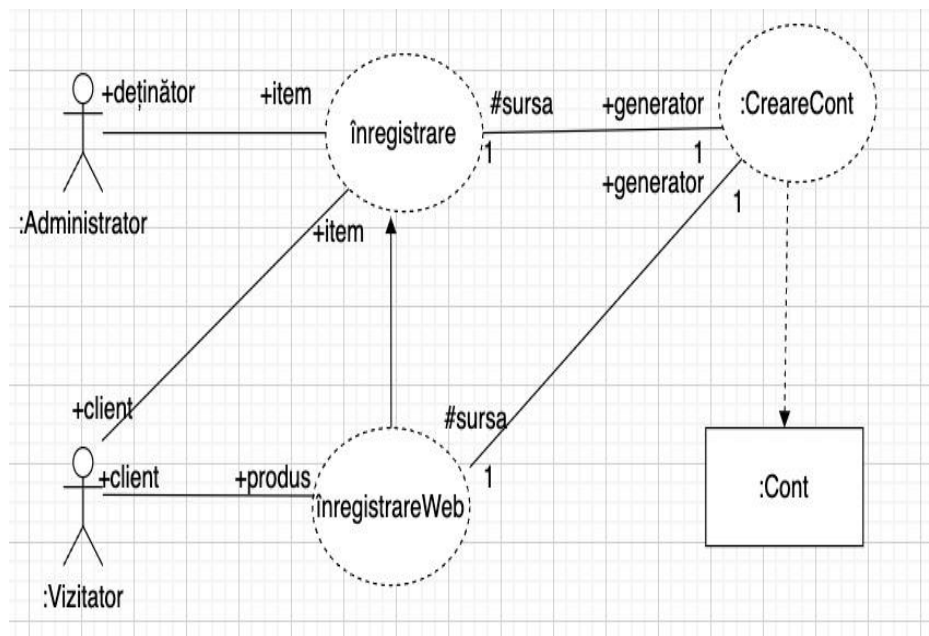


Figura 2.1.13 - Fluxurile de mesaje în procesul de creare a contului

În cazul în care se optează pentru înregistrarea de către administrator atunci acesta are doar posibilitatea de a crea un nume de utilizator și o parolă care ulterior va fi comunicată de către acesta utilizatorului după ce va fi invocată funcția de crearea a unui cont, responsabilă pentru realizarea unui cont propriu zis conectat la baza de date. Acest lucru este reprezentat prin relațiile de asociere dintre obiectul :Administrator, colaborarea înregistrare și use-case-ul CreareCont, care realizează clasa Cont. În cazul primei relații de asociere :Administrator are rolul public de deținător, iar :înregistrare are rolul public de produs, pe când în cea de a doua relație de asociere colaborarea respectivă are rolul de sursă care are specificatorul protected deoarece nu oricine trebuie să aibă acces la datele obținute de la utilizator. De asemenea și use-case-ul CreareCont are funcția de generator la nivelul protected iar relația de multiplicitate a acestei relații este de 1 : 1, dat fiind faptul că unui set de date de intrare îi poate fi creat un singur cont.

Cea de a doua modalitate este înregistrarea independentă a utilizatorului, care poate opta pentru înregistrarea prin introducerea unui nume de utilizator și a unei parole, parcurgând ulterior aceleași pași ca și în cazul precedent, sau prin intermediul unui alt account deja existent, ceea ce am reprezentat prin relația de asociere dintre :Vizitator și colaborarea :înregistrareWeb, care presupune totalitatea opțiunilor de logare cu un cont deja creat precum cel din facebook, google, gmail, instagram, etc. Aceasta colaborare moștenește colaborarea înregistrare și asemenea acesteia se află în relație de asociere cu use-case-ul CreareCont care realizează clasa Cont. Multiplicitatea acestei relații este de asemenea de 1 : 1, iar rolurile îndeplinite de părțile respective coincid cu cea din exemplul precedent având aceeași multiplicitate și același nivel de acces.

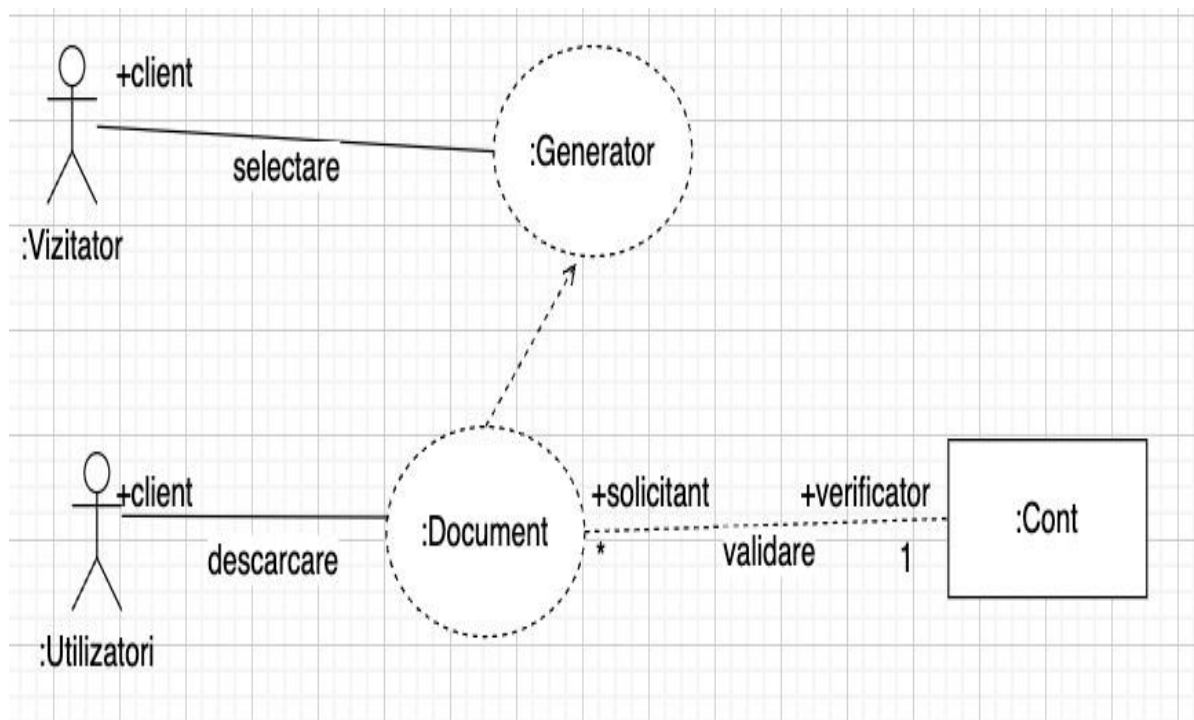


Figura 2.1.14 - Fluxurile de mesaje în procesul de procurare a documentului

Fluxurile de mesaje din cadrul procesului de procurare a testelor este reprezentat în figura 2.1.14. Aici am reprezentat procesul de cumpărare a produselor disponibile, care are ca caz particular descărcarea a documentelor în format electronic. Colaborarea Document reprezintă totalitatea opțiunilor de descărcarea, iar colaborarea Generator reprezintă totalitatea metodelor de cumpărare posibile și permise de către site. Aceste două colaborări se află în relație de dependență colaborarea Document fiind cea dependentă, deoarece fără a finaliza activitatea și a achita produsele nu este posibilă descărcarea acestora, ceea ce înseamnă că modificările din lista de itemi(componentă a colaborării Generator) vor modifica și posibilitățile de descărcare, componente a colaborării Document.

De asemenea posibilitatea de a descărca o vor avea doar utilizatorii care au un cont valid, adică această acțiune poate fi făcută doar de actorii Utilizatori și mai este necesară procedura de validare reprezentată prin relația de asociere denumită validare cu relația de multiplicitate mulți la 1, pentru că pentru un cont sunt permise descărcări multiple după finalizarea plății.

2.2. Descrierea structurală

Pentru a înțelege structura și caracteristicile componentelor platformei de gestionare a testelor de evaluare sumativă și periodic pentru disciplinele cu profil real, este necesară modelarea și analiza următoarelor aspecte:

–descrierea structurii statice a sistemului (Class Diagram)

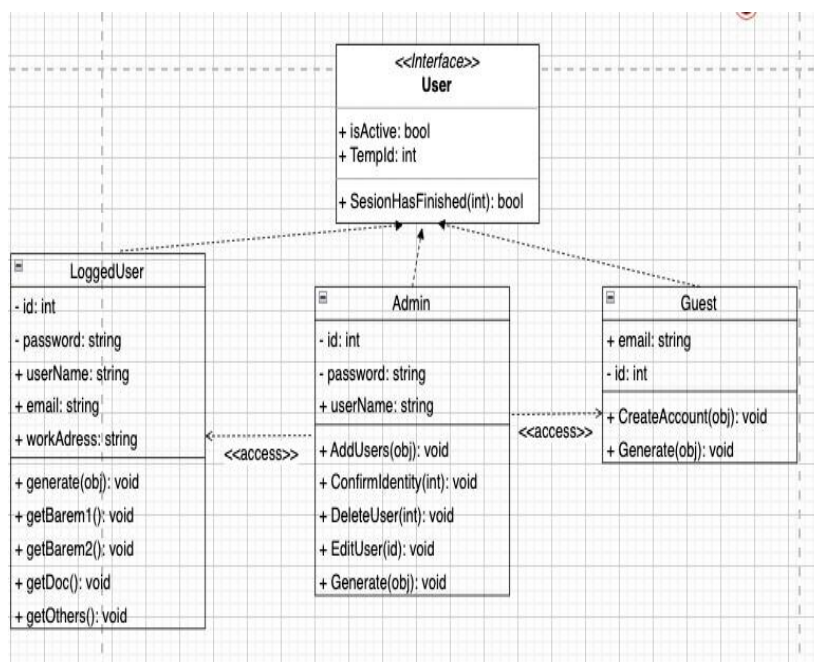


Figura 2.2.1 - Reprezentarea structurală a claselor tipurilor de utilizatori

În diagrama 2.2.1 este ilustrată structura claselor responsabile de autentificarea utilizatorilor. Toate aceste 3 clase realizează interfața User, deoarece indiferent de intenția, inițial necunoscută, a utilizatorului pentru toți se generează un id temporar și se urmărește dacă este încă activ pentru a face posibilă reutilizarea cu scopul de a optimiza memoria utilizată pentru stocarea acestora. Din momentul în care utilizatorul nu a înregistrat nici o activitate în termenul indicat sau dacă a activat înregistrarea, funcția SessionHasFinished returnează valoarea de adevăr și în dependență de context se inițializează procese noi ștergerea Id-ului temporar, crearea unui cont, etc.

După efectuarea înregistrării, utilizatorul va fi ca parte componentă a uneia dintre cele două clase LoggedUser sau Admin. Diferența dintre aceste două constă în faptul că obiectele clasei Admin au accesibilitate la atributele și obiectele clasei Guest și LoggedUser.

- addUsers – oferă posibilitatea de a crea mai multe conturi pentru cadrele didactice cu identitatea cunoscută. Datele necesare pentru logare se trimit ulterior pe poșta cadrelor didactice;
- confirmIdentity – oferă posibilitatea administratorului de a confirma identitatea utilizatorului după verificarea datelor introduse de acesta;
- deleteUser – permite administratorului să șteargă contul unui utilizator cu sau fără salvarea anumitor date despre activitatea utilizatorului respectiv în cadrul platformei;
- editUser – permite administratorului să editeze datele contul unui utilizator cu sau fără salvarea anumitor date despre activitatea utilizatorului respectiv în cadrul platformei, în dependență de dorința utilizatorului și a administratorului.

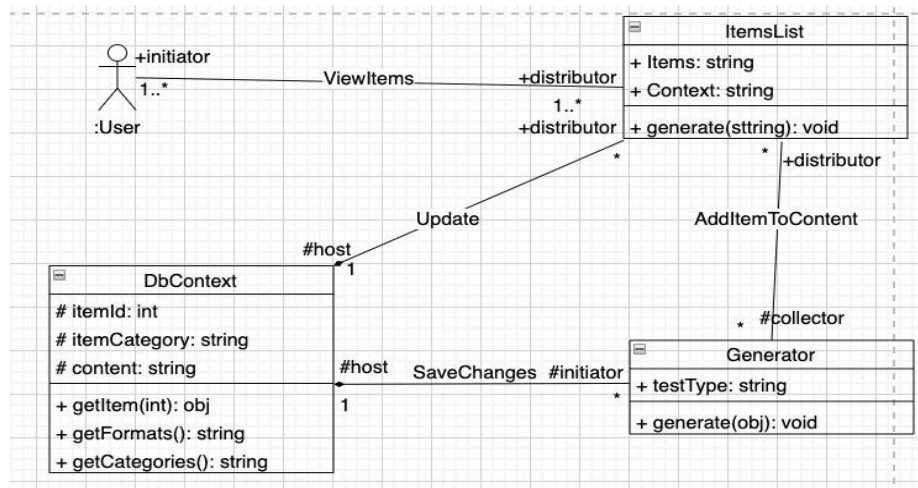


Figura 2.2.2 - Reprezentarea structurală a procesului de adăugare a itemilor în conținutul testului

Conform figurei 2.2.2, utilizatorul inițiază procesul de căutare prin introducerea unui input distinctiv sau declanșarea unui event (prin click), care ulterior este procesat și distribuit de către obiectele clasei ItemsList.

Clasa DbContext interacționează cu clasele Generator și ItemsList între acestea fiind relații de compoziție, partea întreagă fiind Clasa DbContext, deoarece atât generatorul cât și clasa ItemsList nu pot exista independent întrucât distrugerea contextului de bază de date va duce și la distrugerea celor două clase care prin această acțiune vor fi lipsite de conținut (clasa DbContext are rolul de gazdă atât pentru generator cât și pentru itemsList, ceea ce permite realizarea acțiunilor de salvare a schimbărilor și de reînnoire a datelor din sistem).

De asemenea, între clasele generator și itemsList este relație de asociere ca urmare a apelării funcției Generate() din clasa itemslist. Prin urmare itemsList va îndeplini rolul de distribuitor a informației despre itemul distinctiv iar clasa Generator va colecta informația inițializând ulterior un nou proces de salvare a modificărilor în baza de date.

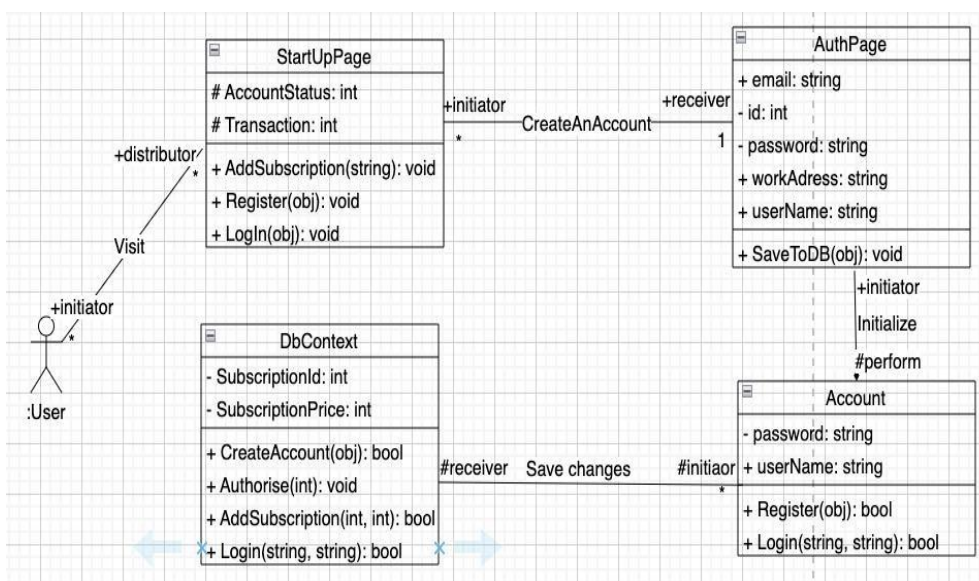


Figura 2.2.3 - Reprezentarea structurală a procesului de înregistrare

Reieșind din figura 2.2.3 utilizatorul vizitează pagina de start de unde poate iniția trei procese:

- a) Login – inițializează procesul de autentificare în baza unui cont de utilizator introdus manual de către administrator în baza de date;
- b) Register – inițializează procesul de înregistrare prin implicarea unei alte clase (AuthPage);
- c) AddSubscription – inițializează procesul de procurare a abonamentului pentru a primi anumite resurse în dependență de tipul de abonament.

Pagina de înregistrare primește de la utilizator fie inputul sau, prin Funcția SelectMethod(), datele de logare de la alte site-uri utilizate (facebook, google, instagram), utilizat pentru inițializarea atributelor sale: email, password și UserName. După care urmează ca utilizatorul să verifice datele de intrare dacă au fost introduse corect și să adauge informațiile despre locul de muncă, după care poate finaliza această acțiune prin declanșarea unui event cu apăsarea unui click. Ca Urmare obiectul clasei RegisterPage va iniția procesul de crearea a unui cont de utilizator, acest proces fiind în continuare realizat de clasa Account care este în relație de compoziție clasa RegisterPage (distrugerea părții întregi va distruge și partea componentă deoarece ambele clase au același scop dar execută anumită părți a acestuia) și de asociere cu clasa DbContext, deoarece ele interacționează pentru a salva modificările în baza de date, aici clasa account având rolul de inițiator iar DbContext de recepționator.

În dependență de informația primită poate fi inițiată una din cele trei funcții:

- a) CreateAccount – creează un cont pentru un nou utilizator;
- b) LogIn – verifică dacă acest cont este existent și oferă informația existentă despre activitatea acestuia;
- c) Authorise – validează identitatea cadrului didactic.

–relațiile de dependență între componentele (Component Diagram)

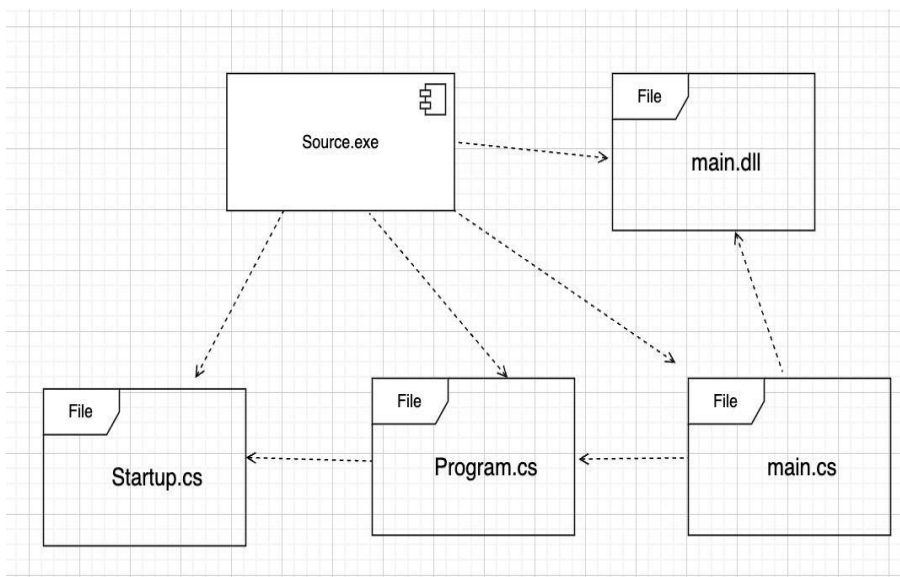


Figura 2.2.4 - Componenta Source.exe din aplicația backend

În diagrama 2.2.4 este reprezentată componenta Source.exe și dependențele ei. Source.exe este componenta dependentă de artefactele main.cs, main.dll, Program.cs și Startup.cs. În același timp artefactul main.cs este dependent de main.dll și Program.cs, iar Program.cs este dependentă de Startup.cs.

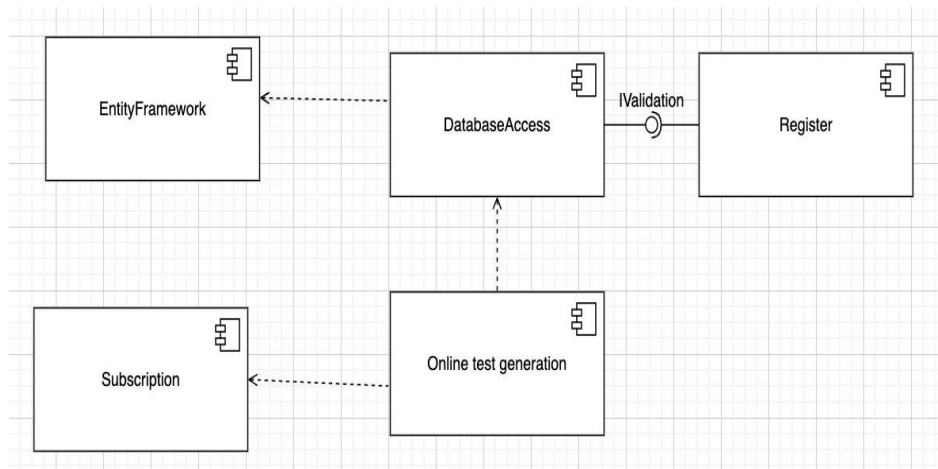


Figura 2.2.5 - Diagrama de componente a platformei

În diagrama 2.2.5 sunt prezentate componentele online test generation (pagina web), databaseAccess, Register și Subscription. Între primele două componente, test generation și databaseAccess este relația de dependență, la fel ca și în cazul componentelor online test generation și Subscription, întrucât modificarea politicii de accesare a datelor sau a tipului abonamentului pentru aplicația respectivă va duce la schimbarea aspectului/ conținutului componentei online test generation. Pe lângă acestea politica de accesibilitate a datelor este dependentă de tehnologia utilizată și anume de EntityFramework. De asemenea există interacțiuni între componentele databaseAccess și Register prin intermediul interfeței IValidation oferită de databaseAccess, necesară componentei Register pentru verificarea datelor de intrare.

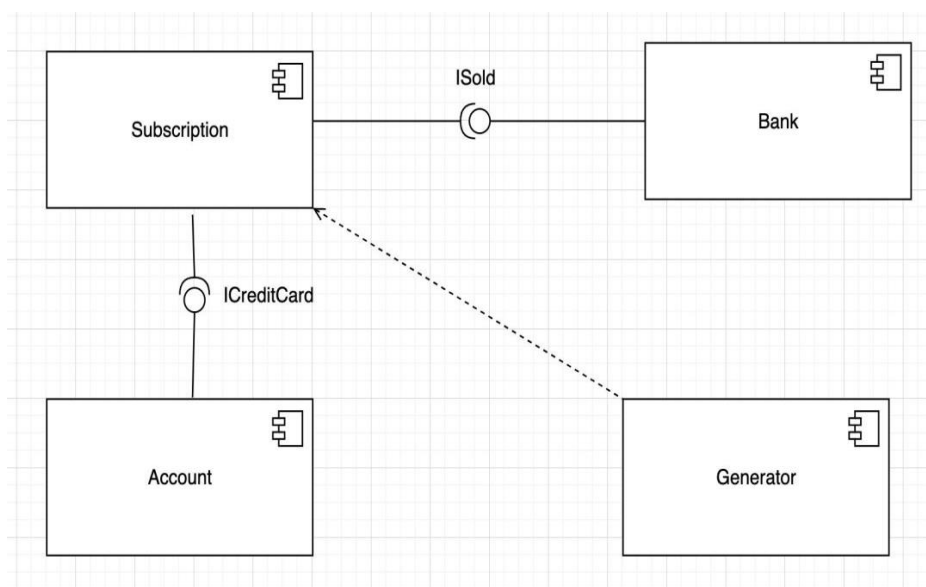


Figura 2.2.6 - Diagrama de componente a procesului de acreditare

În diagrama din figura 2.2.6 sunt prezentate componentele ce participă în procesele de procurare a abonamentului, indiferent de tipul acestuia. Componenta Subscription pentru funcționarea sa are nevoie de interfața ICreditCard oferită de componenta Account și de interfața ISold oferită de componenta Bank. De asemenea cu componenta Generator se află în relație de dependență, deoarece generatorul este dependentă de componenta Subscription.

–modelarea echipamentelor mediului de implementare (Deployment Diagram)

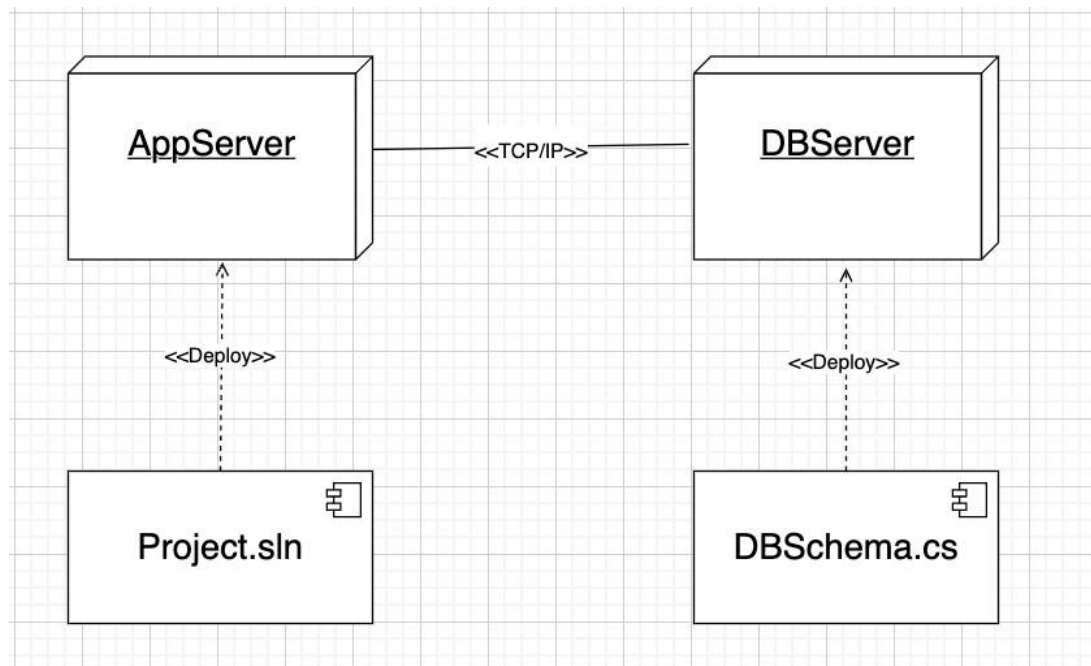


Figura 2.2.7 - Diagrama de plasare.

În figura 2.2.7 este ilustrată digrama de plasare în care sunt două noduri AppServer , care implementează artefactul Project.sln, și DbServer care implementează DBSchema.cs. De asemenea aceste două noduri comunică între ele prin intermediul prin TCP/IP.

3. REALIZAREA SISTEMULUI

Pentru realizarea sistemului dat au fost utilizate următoarele tehnologii și limbaje de programare:

–React - este o bibliotecă populară de JavaScript, open-source, folosită pentru a construi interfețe de utilizator. Dezvoltată și întreținută de către Facebook, React este cunoscută pentru simplitatea, flexibilitatea și performanța ridicată. Unele avantaje ale utilizării React includ:

- a) virtual DOM: React folosește un DOM virtual pentru a actualiza eficient interfața utilizatorului, reducând numărul efectiv de manipulări ale DOM-ului și îmbunătățind performanța;
- b) arhitectură bazată pe componente: Arhitectura bazată pe componente a React permite dezvoltatorilor să construiască componente de interfață utilizator reutilizabile care pot fi ușor distribuite în diferite părți ale unei aplicații sau între diferite aplicații;
- c) sintaxa JSX: Utilizarea sintaxei JSX în React permite dezvoltatorilor să scrie cod similar HTML direct în fișierele lor JavaScript, ceea ce face mai ușor de înțeles și întreținut codul;
- d) ecosistem vast: React are un ecosistem vast și activ de biblioteci, instrumente și extensii care pot ajuta dezvoltatorii să construiască și să gestioneze aplicații complexe mai eficient;
- e) suport pentru testare: React oferă suport robust pentru testare, inclusiv instrumente de testare încorporate și capacitatea de a se integra ușor cu cadrele de testare populare.

–Fetch Api - API Fetch este o API web încorporat în browserele moderne care vă permite să efectuați solicitări HTTP folosind o sintaxă simplă și intuitivă. Acesta oferă o alternativă modernă la API-ul XMLHttpRequest (XHR) pentru a efectua solicitări de rețea în JavaScript. Unele avantaje ale utilizării API-ului Fetch includ:

- a) sintaxă simplă și intuitivă: API-ul Fetch folosește o sintaxă simplă și intuitivă pentru a efectua solicitări HTTP, făcându-l mai ușor de utilizat decât API-ul XHR;
- b) Bazat pe promisiuni: API-ul Fetch se bazează pe promisiuni, ceea ce înseamnă că returnează un obiect Promisiune care se rezolvă cu răspunsul la solicitare. Acest lucru face mai ușor de gestionat codul asincron și evită necesitatea de a folosi funcții de apel înapoi;
- c) Suport încorporat pentru JSON: API-ul Fetch are suport încorporat pentru lucru cu date JSON, inclusiv capacitatea de a analiza automat răspunsurile JSON;
- d) Solicitări personalizabile: API-ul Fetch vă permite să personalizați solicitările HTTP adăugând antete, stabilind metode de solicitare și multe altele.

–MathJax este un motor de afișare realizat în JavaScript pentru ecuații și expresii matematice care funcționează în toate browserele. Acesta oferă instrumente ce face posibilă transformarea conținutului din surse tradiționale de tipărire în conținut web modern și accesibil și ePub. Modulul respectiv este extrem de flexibil și poate fi adaptat nevoilor oricărui utilizator prin crearea de configurații personalizate și fluxuri de lucru software specializate. Mecanismul acestuia folosește CSS cu fonturi web sau SVG, în loc de imagini bitmap sau Flash, astfel încât ecuațiile se scalează cu textul din jur la

toate nivelurile de zoom. Ca urmare generează rezultate de înaltă calitate pe toate browserele și platformele - chiar și browserele vechi, cum ar fi IE 6 (dacă chiar trebuie). Pe lângă acestea este foarte modular la intrare și la ieșire, astfel încât permite utilizarea MathML, TeX și ASCIImath ca intrare și produce HTML+CSS, SVG sau MathML ca ieșire. De asemenea oferă spre utilizare API-urile extinse pentru a crea conținut interactiv, instrumente avansate de creație și aplicații web și mobile activate pentru matematică, ce simplifică esențial efortul necesar pentru integrarea și utilizarea acestui modul în orice proiect.

–Firebase - este o platformă dezvoltată de Google care oferă o serie de servicii și instrumente pentru dezvoltarea rapidă de aplicații mobile și web. Printre acestea se numără un sistem de autentificare, bază de date în timp real, stocare de fișiere, hosting, servicii de notificare și multe altele. Firebase utilizează o arhitectură client-server pentru a permite dezvoltatorilor să își construiască aplicații scalabile și fiabile. Această arhitectură se bazează pe două componente principale:

- a) clientul: Acesta este componenta care rulează în browserul sau dispozitivul utilizatorului și care interacționează cu serviciile Firebase prin intermediul SDK-ului Firebase;
- b) serverul: Acesta este componenta care rulează în cloud-ul Firebase și care asigură stocarea datelor, autentificarea utilizatorilor și livrarea de notificări.

Pentru a utiliza Firebase, dezvoltatorii trebuie să creeze un cont Firebase și să configureze serviciile pe care doresc să le utilizeze în aplicația lor. După configurare, aceștia trebuie să adauge SDK-ul Firebase în codul aplicației lor, astfel încât aceasta să poată comunica cu serviciile Firebase.

SDK-ul Firebase este disponibil pentru o gamă largă de platforme, inclusiv iOS, Android, web și backend. Aceste SDK-uri oferă o serie de metode și funcții pentru a interacționa cu serviciile Firebase și pentru a integra aceste servicii în diverse aplicații.

–Firestore - este un serviciu de baze de date în timp real și scalabile dezvoltat de Google ca parte a platformei Firebase. Aceasta oferă dezvoltatorilor un sistem de stocare de date flexibil și scalabil, care poate fi utilizat pentru a dezvolta o varietate de aplicații mobile și web. Avantajele utilizării acestuia constituie:

- a) scalabilitate: Firestore este un serviciu scalabil care poate crește și se adaptează la necesitățile aplicației dvs. Astfel, este ideal pentru aplicațiile care se așteaptă la o creștere rapidă a traficului sau a datelor stocate;
- b) bază de date în timp real: Firestore permite schimbul de date în timp real între aplicația dvs. și baza de date. Aceasta înseamnă că orice modificare a datelor din baza de date va fi reflectată în timp real în aplicație;
- c) structură de colecții și documente: Firestore utilizează o structură de colecții și documente care este ușor de înțeles și utilizat. Acest lucru face mai ușor pentru dezvoltatori să organizeze și să acceseze datele;

d) interacțiunea bidirecțională: Firestore permite interacțiunea bidirecțională între aplicația dvs. și baza de date, ceea ce înseamnă că puteți efectua operații de citire și scriere atât pe server, cât și pe client;

e) accesibilitate: Firestore este ușor de accesat prin intermediul SDK-ului Firebase, care este disponibil pentru o gamă largă de platforme, inclusiv iOS, Android, web și backend.

3.1 Implementarea funcționalităților de bază

Funcționalitățile de bază a platformei sunt crearea conturilor cu diferite roluri, generarea testelor în funcție de selecție și vizualizarea și salvarea documentelor. De asemenea este important ca funcționalitățile de editare și ștergere precum și de adăugare să nu fie disponibile tuturor utilizatorilor, ci doar în funcție de rolul acestora.

Pentru crearea rutelor se va defini toate rutele și componentele corespunzătoare fiecărei ruta în fișierul principal App.js. Astfel conținutul fișierului App.js este reprezentat în figura 3.1.1. Corespondența dintre ruta dorită și componentul în care se regăsește conținutul și funcționalitățile acesteia se realizează prin modulul de rutare disponibil în React. Cu ajutorul tag-urilor Route specificăm ce component trebuie apelat în funcție de valoarea URL, selectându-se prima opțiune ce corespunde.

```
return (
  <>
  {
    location.pathname !== '/eFizica/printedTest' ? <Header currentUser={currentUser}/> : null
  }
  {
    currentUser ? (
      <Routes>
        <Route path="/" element={<BaseHomePage />}/>
        <Route path="/eFizica" element={<HomePage />}/>
        <Route path="/eFizica/tests" element={<TestPage role={role}/> /> />
        <Route path="/eFizica/lab" element={<LabPage role={role}/> /> />
        <Route path="/eFizica/printedTest" element={<PrintedTest /> /> />
        <Route path="/eFizica/plan" element={<PlanPage role={role}/> /> />
        <Route path="/eFizica/calendar" element={<CalendarPage role={role}/> /> />
        <Route path="/eFizica/tables" element={<TablesPage role={role}/> /> />
        <Route path="/eMath" element={<MathHomePage />}/>
        <Route path="/eMath/tests" element={<MathTestPage role={role}/> /> />
        <Route path="/eMath/printedTest" element={<MathPrintedTest /> /> />
        <Route path="/eMath/plan" element={<MathPlanPage role={role}/> /> />
        <Route path="/eMath/tables" element={<MathTablesPage role={role}/> /> />
        <Route path="/eInfo" element={<InfoHomePage />}/>
        <Route path="/eInfo/tests" element={<InfoTestPage role={role}/> /> />
        <Route path="/eInfo/printedTest" element={<InfoPrintedTest /> /> />
        <Route path="/eInfo/plan" element={<InfoPlanPage role={role}/> /> />
        <Route path="/eInfo/tables" element={<InfoTablesPage role={role}/> /> />
      </Routes>
    ) : (
      <Routes>
        <Route path="/" element={<SHomePage />}/>
      </Routes>
    )
  }
)
```

Figura 3.1.1 – Definirea rutelor disponibile în aplicația web

După crearea rutelor, se vor crea componentele necesare pentru elaborarea paginilor de exemplu header-ul, fișierul căruia va fi de tip jsx , având conținutul din figura 3.1.2.

```
<Typography
  variant="h6"
  noWrap
  component="div"
  sx={{ flexGrow: 1, display: { xs: 'flex', md: 'none' } }}
>
  <Link style={{ color: 'white' }} to={"/"}>eSchool</Link>
</Typography>
<Box sx={{ flexGrow: 1, display: { xs: 'none', md: 'flex' } }}>
  {pages.map((page : {location: string, name: string} ) => (
    <Button
      key={page.name}
      onClick={handleCloseNavMenu}
      sx={{ my: 2, color: 'white', display: 'block' }}
    >
      <Link style={{ color: 'white' }} to={page.location}>{page.name}</Link>
    </Button>
  ))}
  <div style={{position: "absolute", top: 15, right: 5}}>
    {
      currentUser ? (
        <div>
          <Button style={{color: 'white'}} onClick={() => auth.signOut()}>Deconectează-mă</Button>
        </div>
      ) : (
        <div>
          <Button style={{color: 'white'}} onClick={() => {handleOpenSignIn(!openSignIn)}}>Conectare</Button>
          <Button style={{color: 'white'}} onClick={() => {handleOpen(!open)}}>Înregistrare</Button>
        </div>
      )
    }
  </div>
```

Figura 3.1.2 – Fragment din fișierul Header.jsx

Acest component va cuprinde titlul aplicației și opțiunile de înregistrare și autentificare sau deconectare dacă utilizatorul deja a intrat în contul său. De asemenea în header se va regăsi și un meniu orizontal ce va cuprinde redirecționările către paginile de bază a aplicației web, acestea fiind ilustrate în figura 3.1.3.



Figura 3.1.3 – Header-ul aplicației web

După crearea componentelor comune pentru toate paginile, precum header-ul, footer-ului și ferestrei modale, vor fi implementate fișierele individuale specifice pentru fiecare disciplină, ce vor fi utilizate doar în anumite pagini, pentru afișarea conținutului didactic. Astfel pentru fiecare tip de item va fi creat un component separat pentru a putea afișa corespunzător informația acestui tip de item. Prin urmare pentru disciplina fizica vom avea itemi de tip adevărat și fals, itemi de tip corespondență, itemi de completare și itemi cu probleme propuse spre rezolvare, care se vor diviza în două categorii în dependență de prezența sau absența imaginii în condiția problemei.

Pentru itemii de tip corespondență și adevărat sau fals se va crea un card divizat în două coloane. În cazul itemilor de tip corespondență în prima coloană vor fi ilustrate mărimile iar în a doua unitățile

de măsură, pe când în cazul itemilor de tip adevărat sau fals prima coloană va încorpora conținutul itemului, iar a doua opțiunile pentru selecție A și F, după cum este ilustrat și în figura 3.1.4.

```
export const BooleanItem = ({item}) => {
  return (
    <div style={{display: 'flex', flexDirection: 'row', paddingBottom: 10}}>
      <div style={{width: '90%'}}>{item.condition}</div>
      <div style={{width: '10%', textAlign: 'center'}}> A F </div>
    </div>
  )
}
```

Figura 3.1.4 – Implementarea itemilor de tip adevărat sau fals

Itemii de tip completare și problemele fără imagini vor avea o structură asemănătoare fiind alcătuiți ambii din câte o casetă ce nu va fi divizată, ci doar va conține condiția itemului, în cazul itemilor de completare simbolul specific din condiție înlocuind-se cu spațiu pentru completare iar în cazul itemilor cu probleme fără imagini, după condiție va fi adăugat spațiu pentru rezolvarea problemei.

Ultimul tip de item, atât după implementare cât și după utilizare, este itemul pentru problemele ce au imagine în condiție. Pentru crearea și ilustrarea adecvată a acestui tip de item caseta va fi divizată în două rânduri. Primul rând va conține textul condiției iar al doilea rând va fi divizat în două coloane. Prima coloana va conține spațiul pentru rezolvare, iar a doua va conține imaginea din condiția problemei, implementarea aceasta fiind ilustrată în figura 3.1.5.

```
export const SecondProblemItem = ({item}) => {
  const condition = item.condition[0].split('@');
  return (
    <MathJax.Provider>
      <div style={{flexDirection: 'column'}}>
        <p>
          {
            condition.map((element, index) => index % 2 === 0 ? element : (<MathJax.Node key={index} inline formula={element} />))
          }
        </p>
        <img
          style={{width: `${item.condition[2]}`, height: 'auto', paddingBottom: 5, float: 'right', maxHeight: '350px'}} src={item.condition[1]}>
          {
            ...Array(Number(item.condition[3])),
            .map((value, index : number) => (
              <div style={{height: 15}} id={index + 1} key={index}/>
            ))
          }
        </div>
      </MathJax.Provider>
    )
}
```

Figura 3.1.5 – Implementarea itemilor de tip probleme ce conțin imagine

După ce au fost create componentele pentru afișarea corespunzătoare a conținutului din baza de date este necesară implementarea operațiunilor de citire, scriere, editare și ștergere a conținutului didactic din baza de date. Pentru aceasta vor fi create funcții diferite responsabile pentru fiecare din aceste acțiuni, după ce va fi realizată conectarea la firestore.

Pentru utilizatorii cu rol de administrator va fi accesibilă adăugarea conținutului în baza de date, iar pentru asta a fost creată funcția createQuestion ce primește ca parametri atât conținutul itemului cât și disciplina și clasa la care acesta se referă. Implementarea mai detaliată a aceste funcții este ilustrată în figura 3.1.6, de unde putem observa că în funcție de valorile primite se generează calea către

documentul în care se face înserarea.

```
export const createQuestion = async ( grade, module, {...question }, discipline :string = 'physics') => {
  if(!question && !grade && !module) return;

  const questionRef = firestore.collection( collectionPath: `${discipline}/${grade}/${module}`);
  const createdAt = new Date();

  try {
    await questionRef.add({
      createdAt,
      ...question
    })
  } catch (error) {
    console.error(error);
  }

  return;
}
```

Figura 3.1.6 – Implementarea funcției de creare a itemilor

Pentru a putea afișa conținutul a fost creată funcție `getQuestion`, într-un mod similar cu principiul aplicat în cadrul implementării funcției pentru crearea conținutului. Astfel această funcție primește ca argumente atât clasa cât și disciplina și modulul la care se referă itemii doriți. După primirea conținutului din baza de date se va crea un obiect ce va conține itemii sub formă de o listă, precum și clasa, modulul și disciplina relevante pentru itemii respectivi, după cum este ilustrat în figura 3.1.7.

```
export const getQuestion = async ( grade, module, discipline :string = 'physics' ) => {
  if(!grade && !module) return;

  const itemsArr = [];

  const questionRef = firestore.collection( collectionPath: `${discipline}/${grade}/${module}`);
  const querySnapshot = await questionRef.get();

  querySnapshot.docs.forEach(item=>{
    itemsArr.push(item.data())
  })

  return [{
    "discipline": discipline,
    "grade": grade,
    "module": module,
    "questions": itemsArr,
  }];
};
```

Figura 3.1.7 – Implementarea funcției pentru citirea conținutului

După același mecanism au fost create și funcțiile pentru modificarea și ștergerea itemilor din baza de date. Aceste funcții vor fi accesibile doar pentru utilizatorii cu rol de administrator.

3.2 Testarea sistemului

Testarea aplicației web se va face prin testarea manuală, ce va începe cu crearea unui cont de

utilizator cu rol de profesor pentru a putea realiza un test împreună cu anexele necesare. Astfel utilizatorul va completa toate câmpurile din fereastra modală, iar mai exact numele, prenumele, poșta electronică, instituția unde își desfășoară activitatea de bază, titlul pe care îl are și va selecta rolul de profesor sau de student.

După crearea cu succes a unui cont de utilizator al aplicației respective, utilizatorul va fi direcționat către pagina de start a aplicației, unde prin intermediul cardurilor sunt ilustrate toate disciplinele disponibile la moment: matematica și fizica. Pentru testare va fi selectată disciplina fizica.

Pentru a încerca să generăm un test de evaluare, va fi selectată opțiunea de generare a unui test ce va direcționa utilizatorul către pagina de selecție, în care secvențial vor fi ilustrați toți itemii din baza de date în funcție de tipul exercițiului. Aspectul aceste pagini este ilustrat în figura 3.2.1.

Figura 3.2.1 – Pagina de generare a testelor pentru disciplina fizica

Utilizatorul va avea posibilitatea să selecteze modulul dorit pentru crearea testului, precum și clasa, iar în dependență de selecția utilizatorului itemii se modifică pentru a corespunde cu noile valori selectate. În figura 3.2.1 de asemenea pot fi observate iconițele de la fiecare card, acestea fiind vizibile doar utilizatorilor cu cont de administrator, oferindu-le acestora posibilitatea de a edita sau șterge unitățile de conținut din baza de date. De asemenea butonul de adăugare a itemilor este disponibil doar în cazul în care utilizatorul are permisiunile de administrator.

La alegere pentru disciplina fizica aici sunt disponibile 5 module și 7 clase, iar vizualizarea itemilor se va efectua în dependență de tipul itemilor în timp ce parcurgerea poate fi realizată doar prin selectarea numărului corespunzător de itemi. Prin selectare se modifică stilurile componentelor selectate pentru a putea fi accentuate de ceilalți itemi.

Pentru crearea unui test ce corespunde cerințelor curriculare este necesar ca utilizatorul să selecteze un anumit număr de itemi de fiecare tip, prin urmare utilizatorul nu poate trece la etapa de generare dacă nu selectează numărul de itemi indicat.

După selectarea numărului necesar de itemi apare butonul ce permite navigarea către următorul tip

de itemi unde iarăși utilizatorul va avea o serie de itemi disponibili pentru a completa conținutul testului, acest lucru este ilustrat în figura 3.2.2.

Figura 3.2.2 – Parcurgerea secvențială a itemilor

În figura 3.2.2, se observă modificarea stilurilor ce permit cu ușurință să observăm conținutul selectat, iar întrucât selecția corespunde cu cerințele stipulate, și anume cu numărul indicat de itemi de tip completare în cazul dat, apare butonul de navigare ce ne permite să trecem la urătoarea secvență de itemi.

După apăsarea butonului are loc afișarea următorului tip de item, pentru care utilizatorul trebuie să selecteze numărul prestabilit de sub-itemi. Aceste etape sunt parcurse pentru fiecare tip de itemi pentru a genera diferite teste de evaluare. În funcție de testul de evaluare dorit, utilizatorul trebuie să selecteze:

- pentru test fulger, ilustrat în figura 3.2.3 – 3 itemi de completare și 3 itemi de corespondență;
- pentru test sumativ de 45 de minute – 3 itemi de completare, 3 itemi de corespondență, 3 itemi de adevărat și fals și 2 probleme cu grad scăzut de dificultate;
- pentru test sumativ de 60 de minute – 3 itemi de completare, 3 itemi de corespondență, 3 itemi de adevărat și fals și 2 probleme cu grad scăzut de dificultate și o problemă cu grad înalt de dificultate.

Disciplina: Fizica Clasa: 7

Nume, prenume elev: _____ Punctaj acumulat: _____ Nota: _____ Punctaj total: 6

Nume, prenume profesor: _____ Semnatura: _____

Unitatea: Statica fluidelor

| Nr. | Item | Scor |
|-----|---|-----------------------|
| 1 | <p>Continuați propozițiile astfel încât ele să exprime afirmații corecte.</p> <p>Unitate de măsură a presiunii numită _____ comemorează numele savantului francez Blaise Pascal.</p> <p>Legea lui _____ permite să fie explicată plutirea corpurilor.</p> <p>Rezultatul acțiunii forței este mai pronunțat cu atât, cu cât aria suprafeței este mai _____.</p> | L 0 1 2 3 |
| 2 | <p>Asociați mărimile fizice cu unitățile lor (uniți prin săgeți mărimea fizică cu unitatea corespunzătoare ei).</p> <p>Accelerația gravitațională <i>km</i></p> <p>Masa <i>cg</i></p> <p>Înălțimea <i>N/kg</i></p> | L 0 1 2 3 |

Figura 3.2.3 – testul fulger generat

4. DOCUMENTAREA PRODUSULUI REALIZAT

Pentru a atinge obiectivele propuse este necesar, crearea unei documentații la fel de simple și clare ca și produsul în sine. Ca documentație vor fi create două fișiere, primul încorporând toate comenzile disponibile pentru utilizarea codului realizat, iar cel de-al doilea va ilustra toate paginile disponibile împreună cu toate opțiunile accesibile din paginile respective pentru fiecare utilizator în parte.

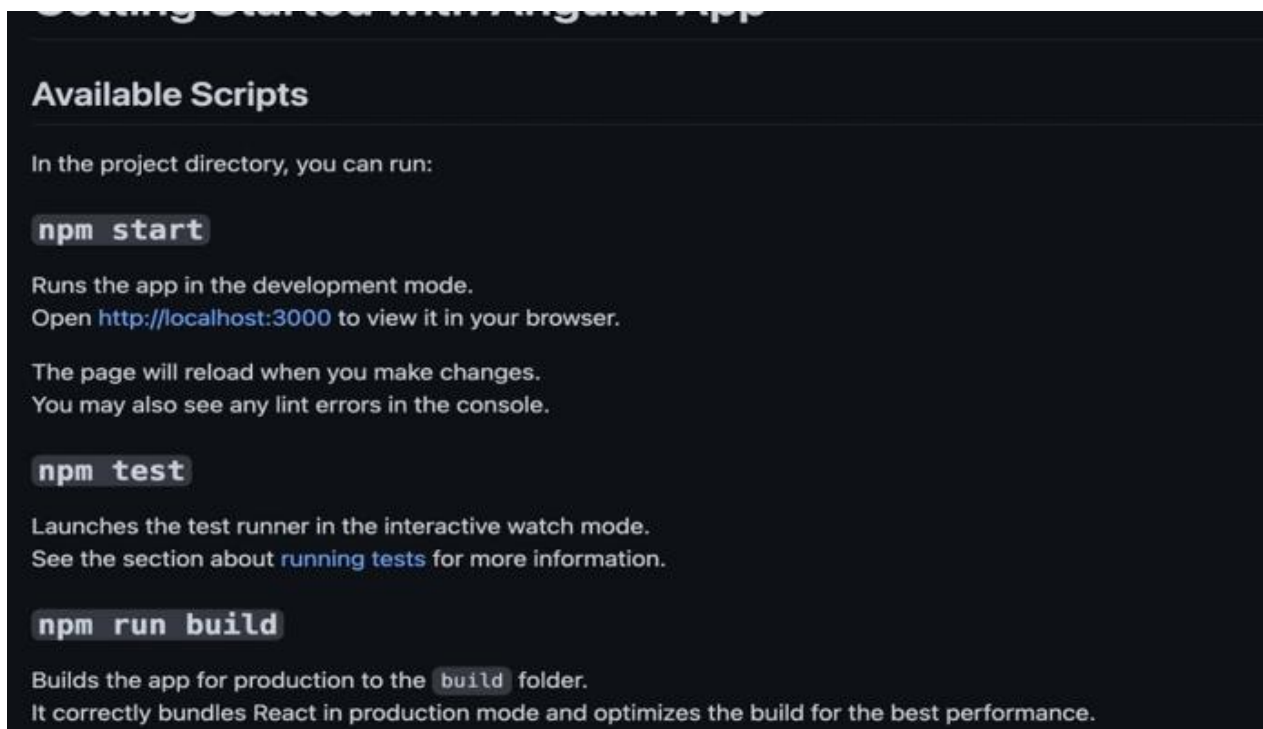


Figura 4.1 – Secvență din fișierul Readme.md

Primul fișier va fi un fișier de tip markdown, conținutul căreia este ilustrat în figura 4.1. Acesta cuprinde toate comenzile disponibile ce se rulează din terminal, cum se scriu corect și ce efecte au acestea. De asemenea este indicat și anumite surse ce vin în ajutor în cazul în care apar anumite neclarități. Acest fișier este destinat persoanelor care vor opera cu codul sursă, la baza fișierului fiind informația necesară pentru rularea proiectului, testarea acestuia și efectuarea modificărilor dacă apare dorința sau necesitatea.

Fișierul respectiv este fișierul pe care un utilizator sau dezvoltator îl deschide când accesează depozitul de cod al proiectului, deoarece acesta servește drept introducere în proiect, oferind atât o imagine generală cât și o anumită familiarizare cu scopul și principiul de funcționare a produsului respectiv. De asemenea acest fișier include instrucțiuni despre cum să se configureze proiectul, inclusiv instrucțiuni de instalare pentru orice dependențe și cum să se ruleze proiectul.

Un alt element important este faptul că conține informații despre structura proiectului, inclusiv structura fișierelor și a dosarelor, precum și scopul fiecărui fișier. Aceste informații sunt utile pentru procesul de mentenanță și ajută la păstrarea arhitecturii proiectului.

În cazul în care utilizatorul depistează anumite neregularități în privința produsului respectiv, acesta poate găsi informații de contact și suport unde poate solicita înlăturarea acestora sau pentru

proponeri de îmbunătățire, utilizatorii pot folosi datele din acest fișier.

Al doilea fișier conține imagini cu fiecare pagină din aplicație urmate de către URL fiecăreia, precum și posibilitățile de interacționare ale utilizatorului cu aplicația realizată în dependență de rolul acestuia. Aspectul acestui fișier este demonstrat în figura 4.2.

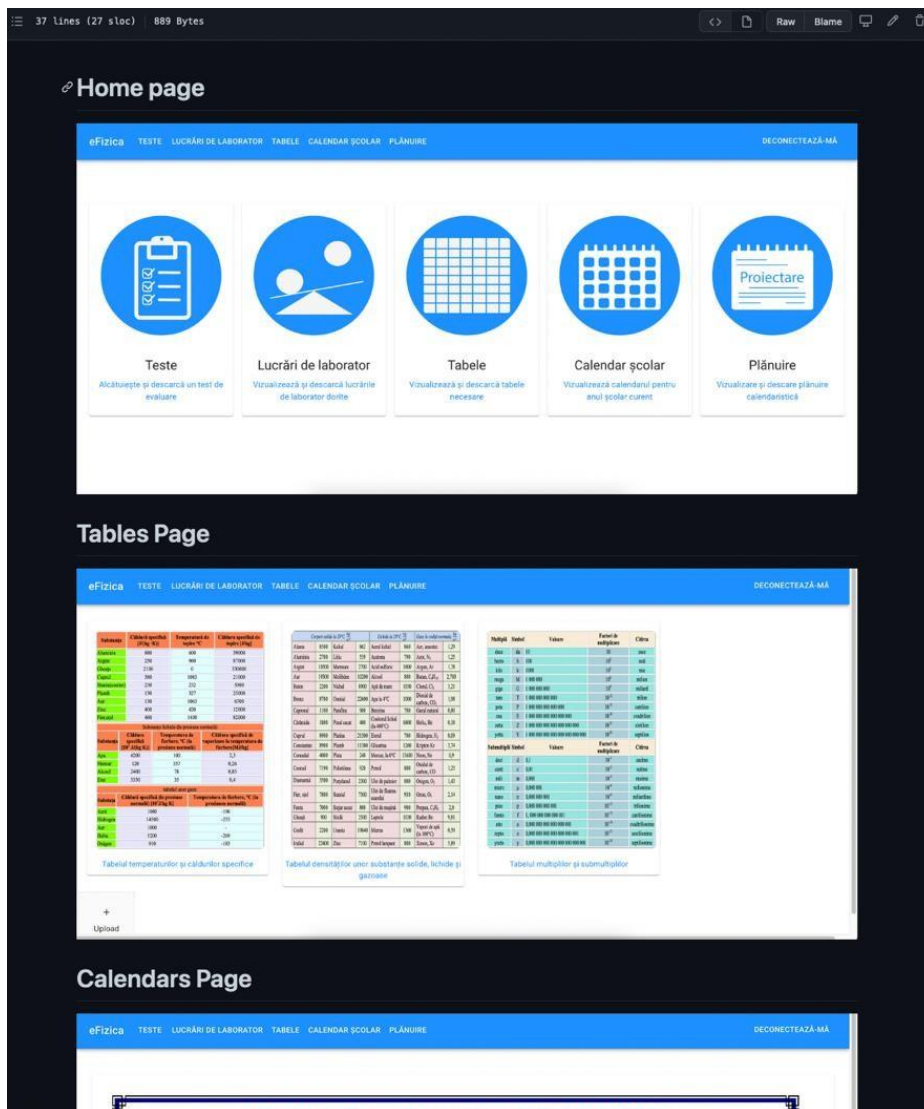


Figura 4.2 – Conținutul fișierului Pages.md

5. ESTIMAREA COSTURILOR PRODUSULUI

Planificarea corectă este esențială în momentul în care dorești să inițiezi o activitate, în special atunci când dorești să creezi o aplicație, deoarece ajută la clarificarea obiectivelor și cerințelor proiectului, definirea amplitudinii proiectului și a calendarului, minimizarea riscurilor și costurilor și îmbunătățirea comunicării și colaborării între membrii echipei. De asemenea un proiect bine planificat are mai multe șanse să fie de succes, să satisfacă nevoile utilizatorilor săi și să fie finalizat la timp și în buget.

Pentru crearea aplicației respective s-au evidențiat câteva faze ce încorporează o serie de activități indispensabile, ilustrate în figura 5.1.

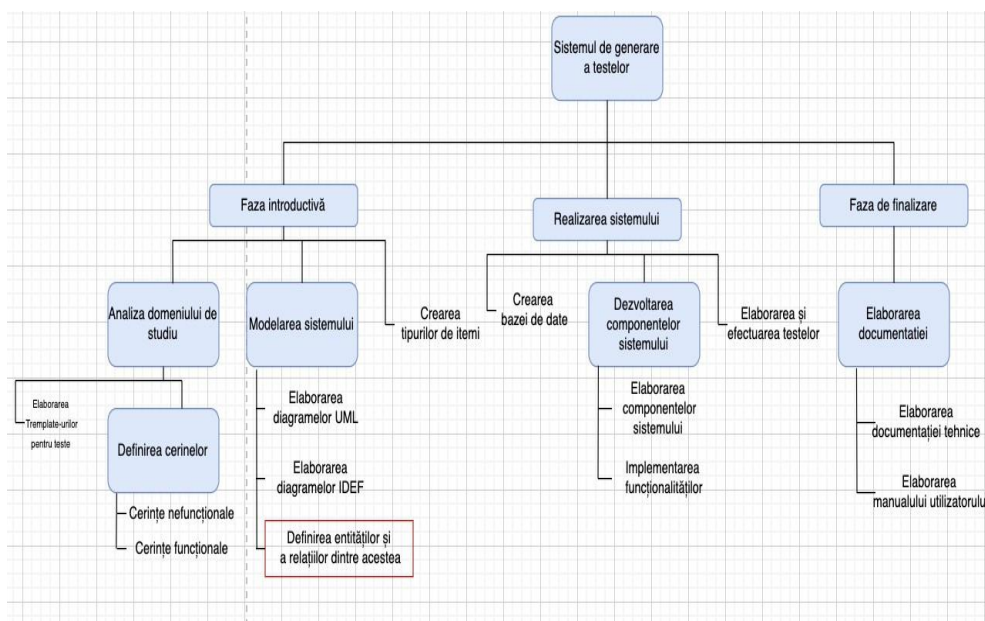


Figura 5.1 – Sarcinile de bază ilustrate prin diagrama WBS

Astfel procesul a început prin crearea unui model ce va permite crearea testelor având la bază doar selecția și modelul respectiv, după care a început definirea cerințelor funcționale și nefuncționale. După aceasta a avut loc modelarea aplicației urmată de definirea entităților ce s-au evidențiat în procesul de modelare. Următoarea este fază de implementare și cea de testare, după care urmează ultima etapă, și anume cea de documentare. Pentru realizarea corectă aplicației este necesar ca etapele să fie parcurse în această ordine, fără a sări/omite anumite etape și activități.

După stabilirea etapelor și activităților necesare de a fi întreprinse pentru realizarea aplicației respective, aproximăm timpul necesar pentru realizarea fiecărei activități în parte. Astfel vom putea aproxima timpul necesar pentru realizarea aplicației și pentru procesul de mentenanță. Deși nu putem determina cu exactitate timpul necesar pentru fiecare etapă dar putem estima în comparație cu celelalte etape. Astfel cea mai costisitoare etapă din punct de vedere a timpului este etapa de implementare. Timpul necesar pentru fiecare etapă este ilustrat în figura 5.2.

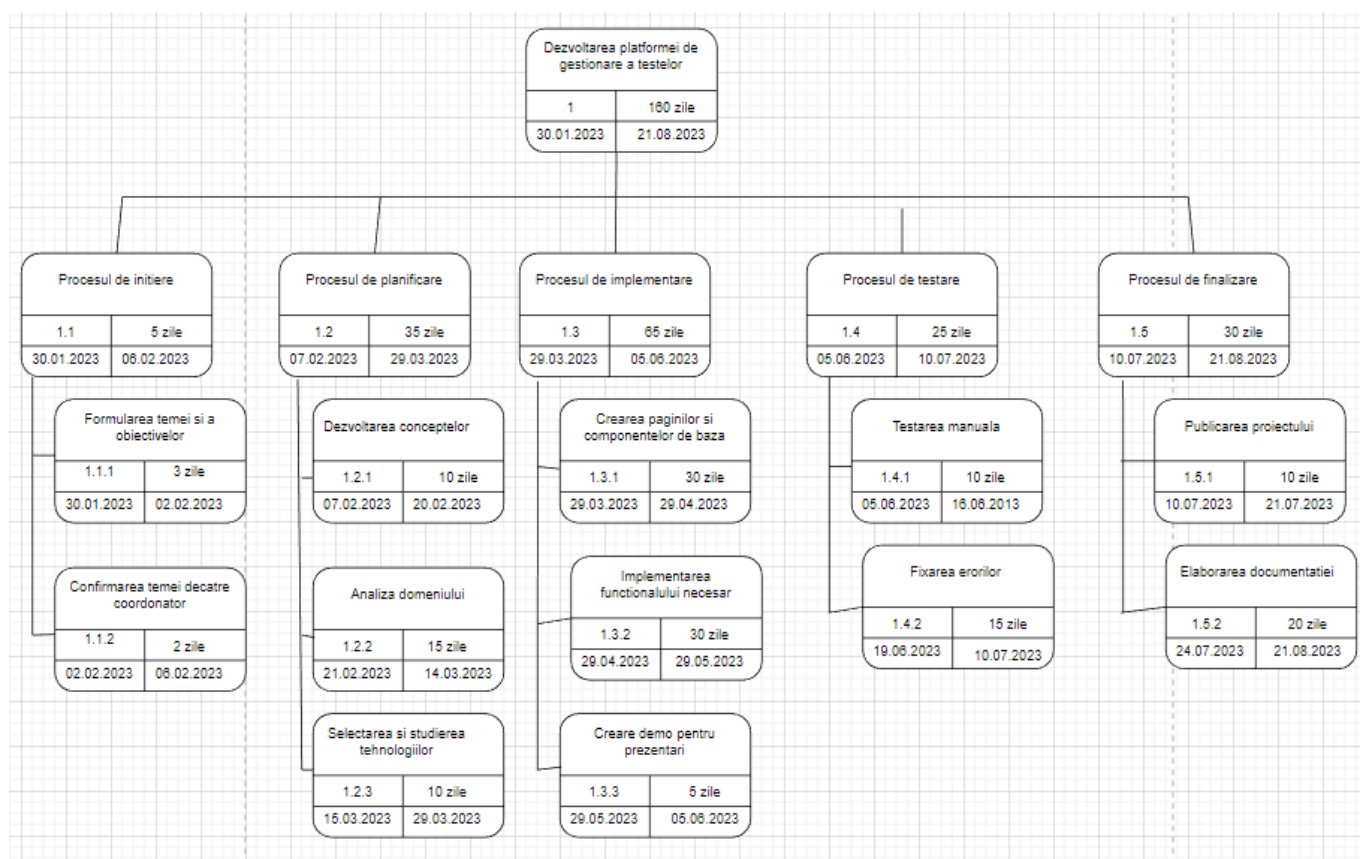


Figura 5.2 – Diagrama repartizării timpului necesar pentru realizarea produsului

Datele din diagrama din figura 5.2 sunt conform celui mai bun scenariu, întrucât reprezintă timpul minimal necesar pentru realizarea platformei de gestionare a testelor de evaluare sumativă pentru disciplinele școlare cu profil real, precum Matematica și Fizica.

Un alt factor important este planificarea costurilor financiare, întrucât cheltuielile ce pot apărea în procesul de dezvoltare vor fi destul de mari. Pentru a estima costurile financiare ale acestui produs vor fi utilizate salariile medii lunare pentru dezvoltatorii software din Republica Moldova la momentul actual și timpul estimat în etapa precedentă.

Astfel pentru un dezvoltator începător din domeniul web, pentru o oră de lucru pe partea de frontend în Republica Moldova se achită în medie până la 200 de lei, iar pe partea de backend se achită până la 250 de lei pe oră. Dev-ops primește între 200 și 250 de lei, iar pentru project manager 300 de lei în timp ce pentru redactarea documentelor se achită până la 80 de lei.

În cazul planificării costurilor vor fi utilizate salariile medii și anume pentru dezvoltatorul frontend plata pentru o oră va fi 140 de lei, pentru backend va fi 180 de lei, iar pentru dev-ops va fi 150 de lei. Pentru project manager vor fi repartizați câte 200 de lei pentru fiecare oră, iar pentru redactarea documentelor va fi 67 de lei / oră. Astfel costul total va fi de 118 710 lei, adică aproximativ 6000 de euro, repartizarea acestora fiind ilustrată în figura 5.3.

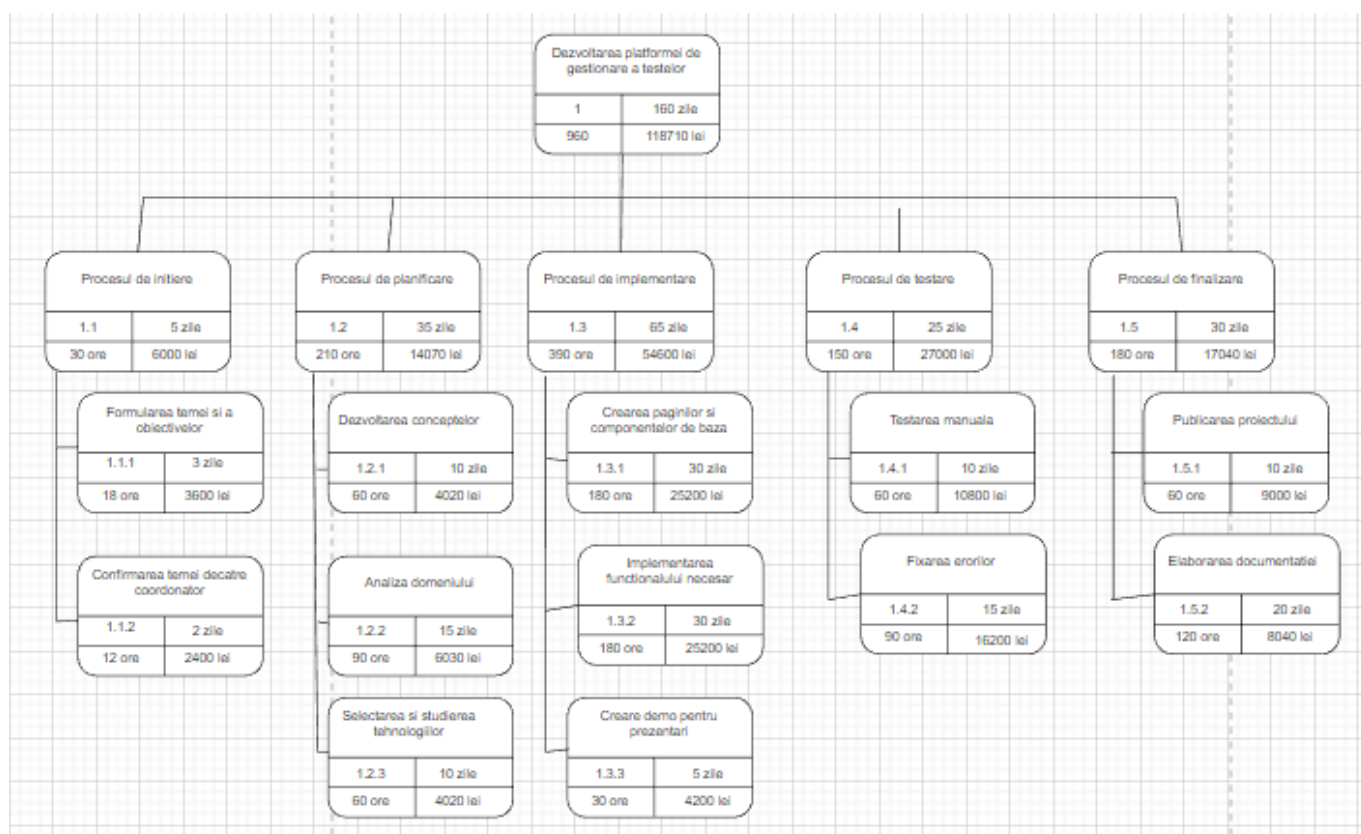


Figura 5.3 – Estimarea costurilor

În urma efectuării tuturor estimărilor, putem evidenția și trăsăturile distinctive ale aplicației respective pentru a evidenția importanța și impactul acesteia. Astfel acest produs nu are concurenți și reprezintă o soluție simplă și eficientă, însă necesită conexiune la internet pentru a funcționa. De asemenea acest produs poate dezvoltat în continuare, ne având concurenți la momentul actual în Republica Moldova. Descrierea mai detaliată este ilustrată în diagrama SWOT din figura 5.4.

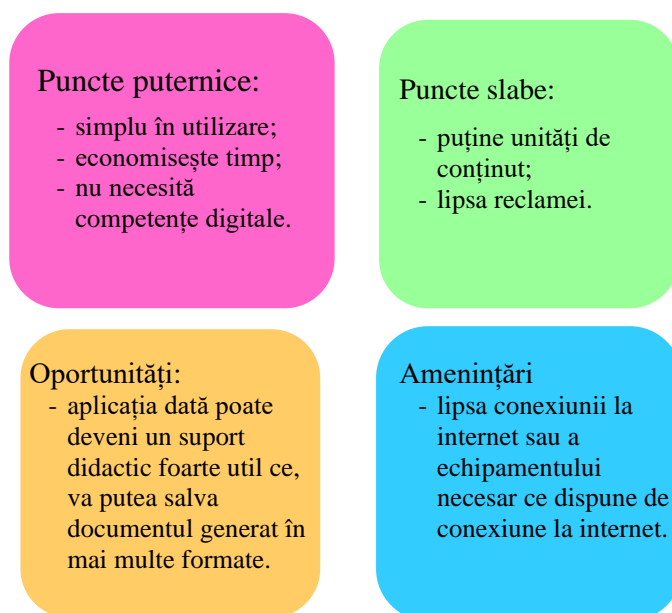


Figura 5.4 – Analiza SWOT

6. CONCLUZII

În urma realizării licenței am creat o aplicație web single-page ce permite cu generarea a 3 tipuri de teste cu eforturi minimale din partea utilizatorului. Aceste teste pot fi printate sau salvate la dorința utilizatorilor.

Prin realizarea acestei aplicații am observat avantajele modulelor utilizate, precum: documentația clară , simplitatea utilizării, resursele minimale necesare pentru implementare, eficiența în utilizare și ușurința cu care pot fi supuse modificărilor.

Astfel am observat că React este un modul extrem de intuitiv, în ceea ce privește modul de lucru și oferă interactivitate la aspectul oricărei interfețe de utilizare. În plus, permite dezvoltarea rapidă și de calitate a aplicațiilor care, la rândul său, economisește volum atât pentru clienți, cât și pentru dezvoltatori.

De asemenea permite efectuarea modificărilor semnificative ale datelor care au ca rezultat modificarea automată a părților selectate ale interfețelor utilizatorilor. Datorită acestei funcționalități progresive, nu există nicio funcție suplimentară pe care trebuie să o efectuați pentru a vă actualiza interfața cu utilizatorul. Acesta oferă componente reutilizabile pe care dezvoltatorii au autoritatea de a le reutiliza și de a crea o nouă aplicație. Reutilizarea este exact ca un remediu pentru dezvoltatori. Acest modul oferă dezvoltatorilor autoritatea de a folosi în caz de necesitate aceleași componentele construite pentru alte aplicații care au aceeași funcționalitate, reducând astfel efortul de dezvoltare și asigurând o performanță impecabilă .

De asemenea pentru realizarea acestei aplicații a fost necesară crearea unui funcțional pentru a putea stoca eficient unitățile de conținut și datele despre utilizatori.

Un alt modul importat este Mathjax, prin integrarea căruia am obținut cu eforturi minimale rezultate foarte bune, având în vedere că ilustrarea adecvată a formulelor și expresiilor matematice reprezintă o sarcină extrem de anevoioasă .

Această aplicație reprezintă un instrument foarte util pentru sistemul educațional, ce permite crearea evaluărilor sumative în timp record fără a necesita anumite cunoștințe pentru crearea și redactarea documentelor.

Pentru profesori această aplicație face posibilă generarea diferitor variante cu ușurință , precum și crearea testelor fulger pentru verificarea cunoștințelor obținuți în urma lecției respective. De asemenea profesorul obține un număr foarte mare de itemi ce pot fi dați elevilor pentru exersare. Forma prestabilită a documentului permite generarea baremului de notare și baremului de corectare cu foarte mare ușurință, economisind timpul și resursele cadrului didactic, ce pot fi direcționate în alte direcții, precum explicarea temelor sau interacțiunea cu elevii.

Pentru elevi acest mecanism de generare a testelor aduce câteva beneficii foarte importante, precum claritatea condițiilor și sarcinilor ce necesită de a fi realizate, forma asemănătoare cu

examenale de stat, ce permite acomodarea cu sistemul de corectare și notare. De asemenea aceștia pot obține numeroase variante pentru exersare.

Importanța acestei aplicații constă în faptul că la elaborarea unui test este nevoie în primul rând de timp și desigur anumite competențe fie digitale fie specifice disciplinei în parte, ceea ce este foarte dificil pentru cadrele didactice ce predau disciplina dată după recalificare venind din alt domeniu sau predarea unei discipline anumite ca disciplină secundară ne fiind specialist. Această aplicație permite obținerea diferitor tipuri de teste în scurt timp indiferent de competențele digitale. Această aplicație își îndeplinește scopul articolului de a ajuta fără excepție toate cadrele didactice în obținerea unui număr mare de evaluări pentru un număr mare de elevi, elaborarea testelor cu grad diferit de complexitate, elaborarea testelor de evaluare inițială, formativă și sumativă .

Organizarea aplicației este una clasică , corectă din punct de vedere științific, clară și accesibilă la utilizare. Astfel, cadrul didactic are acces atât la elaborarea testului cât și la tabelele necesare pentru soluționarea situațiilor semnificative din această aplicație și nu numai. Cadrul didactic are acces la elaborarea testului pe două căi accesând butoanele situate în poziția de sus a paginii sau a ecusonului specific testului. Condițiile de parcurgere a mersului elaborării testului sunt clare, corecte și ușor realizabile atât pentru cadre didactice cât și pentru elevi. Testul poate fi imprimat cât și salvat în dispozitivul individual al cadrului didactic.

Tema analizată este importantă prin prisma faptului că permite economia timpului cadrului didactic la elaborarea testului. Este o aplicație utilă pentru toate cadrele didactice fie cu experiență , tinere, după recalificare, care citesc o disciplină anumită ca ne fiind specialist, etc. De asemenea aplicația oferă o soluție adecvată , reală și gratuită pentru cadrul didactic privitor la o problemă veche, dar care rămâne de actualitate, testul de evaluare electronic.

7. BIBLIOGRAFIE

1. Viorel Bocancea, Victor Ciuvaga, Vladimir Donici, Olga Machevnina ș.a. Curriculum național Fizică clasele VI-IX, Chișinău 2020;
2. Viorel Bocancea, Victor Ciuvaga, Tamara Rusu Ghid de implementarea a curriculumului disciplinar la Fizică, Chișinău 2020;
3. Svetlana Beleaeva, Anatolii Homenco, David Iaroșevici Didactica fizicii: teorie și practică, ProDidactica, Chișinău 2013;
4. ȘTIINȚE EXACTE Matematică, Fizică, Informatică. Curriculum național. Ghid metodologic de implementare pentru învățământul liceal, ProDidactica, Chișinău 2000;
5. Radu I.T., Teorie și practică în evaluarea eficienței învățământului, EDP, București, 1978;
6. Tatiana Șova, Dorina Putină, Evaluarea în învățământ, Tipografia din Bălți 2017;
7. Emanoil Tereja Metodica generală de predare: Fizica, Editura Arc 2001;
8. Bocoș M., Juncan D. Teoria și metodologia instruirii. Teoria și metodologia evaluării. Repere și instrumente didactice pentru formarea profesorilor. București: Paralela 2017;
9. Muraru E. Evaluarea rezultatelor academice. Chișinău: USM, 2004;

8. ANEXE

test fulger.pdf
Page 1 of 2

Disciplina: Fizica Clasa: 7

Nume, prenume elev: _____ Punctaj acumulat: ____ Nota: ____ Punctaj total: 8

Nume, prenume profesor: _____ Semnatura: _____

Unitatea: Mișcarea și repausul

| Nr. | Item | Scor | | | | | | | | | | |
|------------------|---|-----------------------|--------|--------|--------|------------------|-----|--------------|--------|-----------------|--------|---------------------------------|
| 1 | <p>Continuați propozițiile astfel încât ele să exprime afirmații corecte.</p> <p>Mărimea fizică ce caracterizează rapiditatea mișcării mobilului se numește _____.</p> <p>Locul în care se află corpul dat exprimă _____ corpului în spațiu.</p> <p>Lungimea traiectoriei se numește _____.</p> | L 0 1 2 3 | | | | | | | | | | |
| 2 | <p>Asociați mărimile fizice cu unitățile lor (uniți prin săgeți mărimea fizică cu unitatea corespunzătoare ei).</p> <table border="0"> <tr> <td>Aria suprafeței</td> <td>dm^2</td> </tr> <tr> <td>Durata</td> <td>cm/s</td> </tr> <tr> <td>Viteza momentană</td> <td>s</td> </tr> <tr> <td>Viteza medie</td> <td>cm/s</td> </tr> <tr> <td>Viteza corpului</td> <td>cm/s</td> </tr> </table> | Aria suprafeței | dm^2 | Durata | cm/s | Viteza momentană | s | Viteza medie | cm/s | Viteza corpului | cm/s | L 0 1 2 3 4 5 |
| Aria suprafeței | dm^2 | | | | | | | | | | | |
| Durata | cm/s | | | | | | | | | | | |
| Viteza momentană | s | | | | | | | | | | | |
| Viteza medie | cm/s | | | | | | | | | | | |
| Viteza corpului | cm/s | | | | | | | | | | | |

Figura 8.1 – Generarea testului fulger pentru disciplina fizica

test v2.pdf

Page 1 of 2

Search

Disciplina: Fizica

Clasa: 7

Nume, prenume elev: _____

Punctaj acumulat: _____

Nota: _____

Punctaj total: 25

Nume, prenume profesor: _____

Semnatura: _____

Unitatea: Mișcarea și repausul

| Nr. | Item | Scor |
|-----|--|---------------------------------|
| 1 | <p>Continuați propozițiile astfel încât ele să exprime afirmații corecte.</p> <p>Mărimea fizică ce caracterizează rapiditatea mișcării mobilului se numește _____</p> <p>Locul în care se află corpul dat exprimă _____ corpului în spațiu.</p> <p>Lungimea traiectoriei se numește _____.</p> | L 0 1 2 3 |
| 2 | <p>Asociați mărimile fizice cu unitățile lor (uniți prin săgeți mărimea fizică cu unitatea corespunzătoare ei).</p> <p>Aria suprafeței cm/s</p> <p>Durata s</p> <p>Viteza momentană cm/s</p> <p>Viteza medie cm/s</p> <p>Viteza corpului dm³</p> | L 0 1 2 3 4 5 |
| 3 | <p>Determină valoarea de adevăr a următoarelor afirmații, marcând A, dacă afirmația este adevărată, sau F, dacă afirmația este falsă:</p> <p>Deplasarea poate fi mai mare ca drumul parcurs. A F</p> <p>Mărimea fizică ce caracterizează rapiditatea mișcării mobilului se numește viteză. A F</p> <p>La mișcarea rectilinie uniformă viteza corpului se modifică. A F</p> | L 0 1 2 3 |
| 4 | <p>În spațiul rezervat prezentați rezolvările succinte ale itemilor.</p> <p>Care este viteza vântului, dacă o frunză este purtată în 45 minute la o distanță de 5,4 km?</p> | L 0 1 2 3 4 |

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|-----------|--|----------------------------|---|---|---|---|---|---|---|---|---|---|-----------|---|---|---|---|---|---|---|---|---|---|-----------|---|---|---|---|---|---|---|---|---|---|--------------------------------------|
| 5 | <p>În spațiul rezervat prezentați rezolvările succinte ale itemilor.</p> <p>Un melc, în mișcare rectilinie uniformă, parcurge distanța de 30 cm într-un minut. Care este în mm/s, viteza melcului?</p> | L 0 1 2 3 4 | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 6 | <p>În spațiul rezervat prezentați rezolvarea completă a itemului.</p> <p>Două mobile se deplasează din două puncte diferite în aceeași direcție. Aflați momentul și punctul de întâlnire, viteza medie a fiecărui mobil, vitezele în punctele de întâlnire a mobilelor. La calcule se vor lua în considerație datele de mai jos.</p> <table> <tr> <td>$t(h)$</td> <td>0</td> <td>1</td> <td>2</td> <td>3</td> <td>4</td> <td>5</td> <td>6</td> <td>7</td> <td>8</td> <td>9</td> </tr> <tr> <td>$x_1(km)$</td> <td>1</td> <td>2</td> <td>2</td> <td>3</td> <td>4</td> <td>3</td> <td>2</td> <td>1</td> <td>1</td> <td>1</td> </tr> <tr> <td>$x_2(km)$</td> <td>0</td> <td>1</td> <td>1</td> <td>2</td> <td>3</td> <td>4</td> <td>3</td> <td>3</td> <td>2</td> <td>2</td> </tr> </table> | $t(h)$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | $x_1(km)$ | 1 | 2 | 2 | 3 | 4 | 3 | 2 | 1 | 1 | 1 | $x_2(km)$ | 0 | 1 | 1 | 2 | 3 | 4 | 3 | 3 | 2 | 2 | L 0 1 2 3 4 5 6 |
| $t(h)$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | | | | | | | | | | | | | | | | | | | | | | | | | |
| $x_1(km)$ | 1 | 2 | 2 | 3 | 4 | 3 | 2 | 1 | 1 | 1 | | | | | | | | | | | | | | | | | | | | | | | | | |
| $x_2(km)$ | 0 | 1 | 1 | 2 | 3 | 4 | 3 | 3 | 2 | 2 | | | | | | | | | | | | | | | | | | | | | | | | | |

Figura 8.2 – Test de evaluare sumativă pentru 60 de minute pentru disciplina Fizica

