**Proiect Ingineria Programării**

1. **Introducere**

În cadrul acestui proiect va fi prezentat un site web de tip “revistă online” prin prisma celor patru faze fundamentale ale metodologiilor ingineriei programării:

* Analiza (ce s-a dorit construi)‏;
* Proiectarea (cum s-a construit)‏;
* Implementarea (construirea propriu-zisă)‏;
* Testarea (asigurarea calităţii)‏.

1. **Concepția aplicației și a interfeței**

**2.1 Analiza cerințelor beneficiarului**

În faza de analiză, utilizatorul (beneficiarul) a prezentat cerințele sale specifice pentru realizarea produsului dorit. Inițial, acestea au fost următoarele:

* Proiectul va utiliza o baza de date MySQL si va fi programat în PHP.
* Prin intermediul aplicatiei, se vor efectua operatii de stergere, adaugare, citire asupra bazei de date.
* Va exista o pagina de autentificare/înregistrare de utilizatori.
* Vor exista mai multe categorii de utilizatori. Fiecare categorie va avea anumite actiuni specifice.
* Aplicatia va contine mai multe pagini dinamice cu legaturi între ele.
* Va exista posibilitatea de generare si vizualizare de rapoarte (nu doar HTML si PHP).
* Elemente statistice ale site-ului: vizitatori, accesari etc.
* Formular de contact, cu posibilitatea trimiterii de email-uri.

Pentru o înțelegere aprofundată a cerințelor, au avut loc discuții de clarificare, acestea fiind reformulate (acolo unde a fost cazul) astfel încât să fie specifice, măsurabile, accesibile, relevante și disponibile în timp util (SMART). De asemenea, cerințele au fost sortate în ordinea priorităților:

1. Proiectul va utiliza o bază de date MySQL, va fi programat în PHP și va conține cel puțin cinci pagini dinamice cu legături între ele.
2. Prin intermediul aplicației, se vor efectua operatii de stergere, adaugare, citire asupra bazei de date.
3. Va exista posibilitatea autentificării/înregistrării utilizatorilor, cu protecție anti-spam, iar parolele vor fi stocate criptate în baza de date.
4. Vor exista cel puțin două categorii de utilizatori. Fiecare categorie va avea anumite actiuni specifice.
5. Va exista posibilitatea de generare si vizualizare de rapoarte (nu doar HTML si PHP) care să conțină elemente statistice ale site-ului: vizitatori, accesari, vizualizări etc.
6. Va exista un formular de contact, cu posibilitatea trimiterii de email-uri.

Termenul pentru realizarea proiectului a fost de două săptămâni de la transmiterea cerințelor.

Pentru înțelegerea aprofundată a cerinței nr. 4, am întocmit o diagramă UML de tip “caz de utilizare”, care să releve acțiunile pe care le poate face fiecare categorie de utilizator:

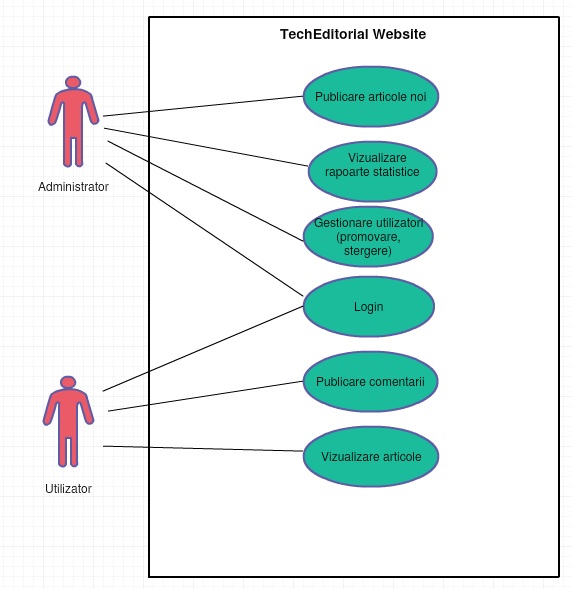


Figura 1: Caz de utilizare administrator și utilizator obișnuit

* 1. **Proiectarea aplicației**

În faza de proiectare, pe baza cerinţelor din faza de analiză, am identificat prioritățile critice:

* Sistem de autentificare/înregistrare utilizatori sigur:
  + Protecție anti-spam – am decis utilizarea Google Captcha, întrucât este o componentă care poate fi reutilizată ulterior pentru formularul de contact;
  + Stocarea criptată a parolelor în baza de date – am decis utilizarea algoritmului MD5 (deși nu este foarte sigur, l-am ales datorită ușurinței de implementare, având în considerare termenul dat)
  + Protecție împotriva injecțiilor SQL a bazei de date – utilizând funcții specifice PHP, am protejat toate formularele de astfel de atacuri.
* Interfață modernă, responsive, care să ofere diferite facilități categoriilor diferite de utilizatori.

Esențială în proiectarea interfeței a fost utilizarea unor tehnologii precum HTML 5, CSS 3 și Bootstrap, care permit realizarea unor componente responsive reutilizabile.

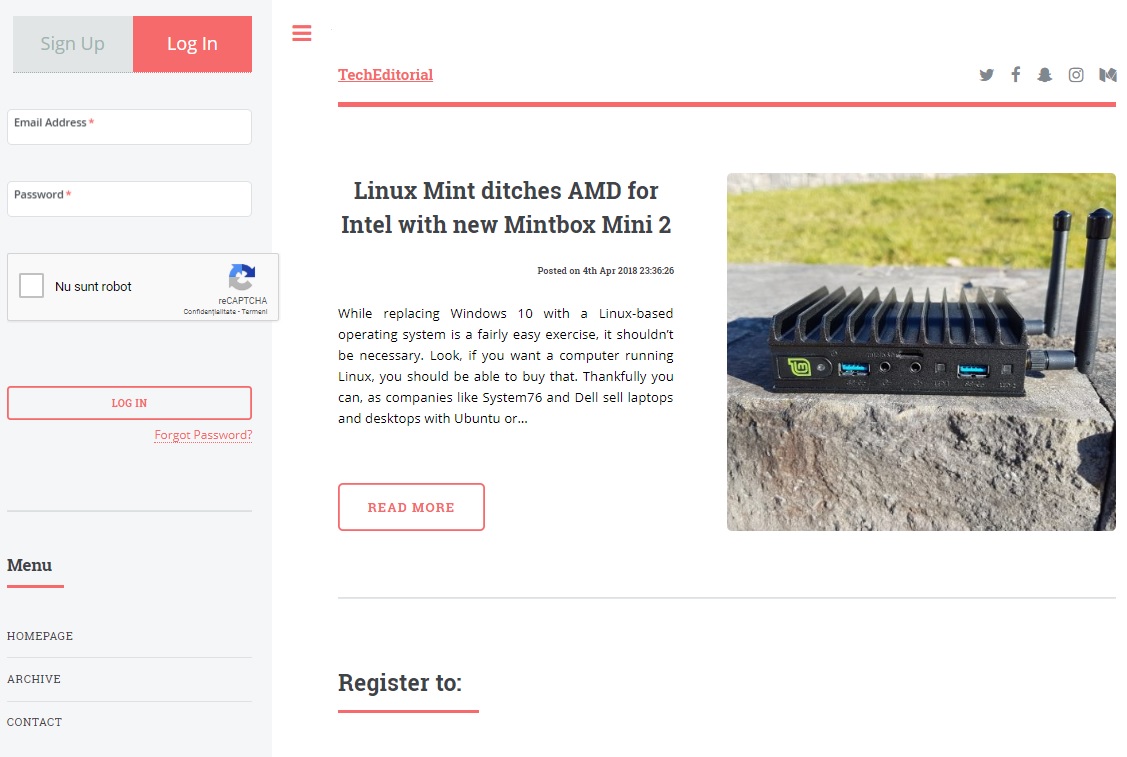


Figura 2: Interfața site-ului web

Pentru stabilirea comportamentului website-ului (răspunsul interfeței la stimulii acţiunilor altor componente, în special ai sistemului de autentificare/înregistrare), am întocmit o diagramă UML de activități:

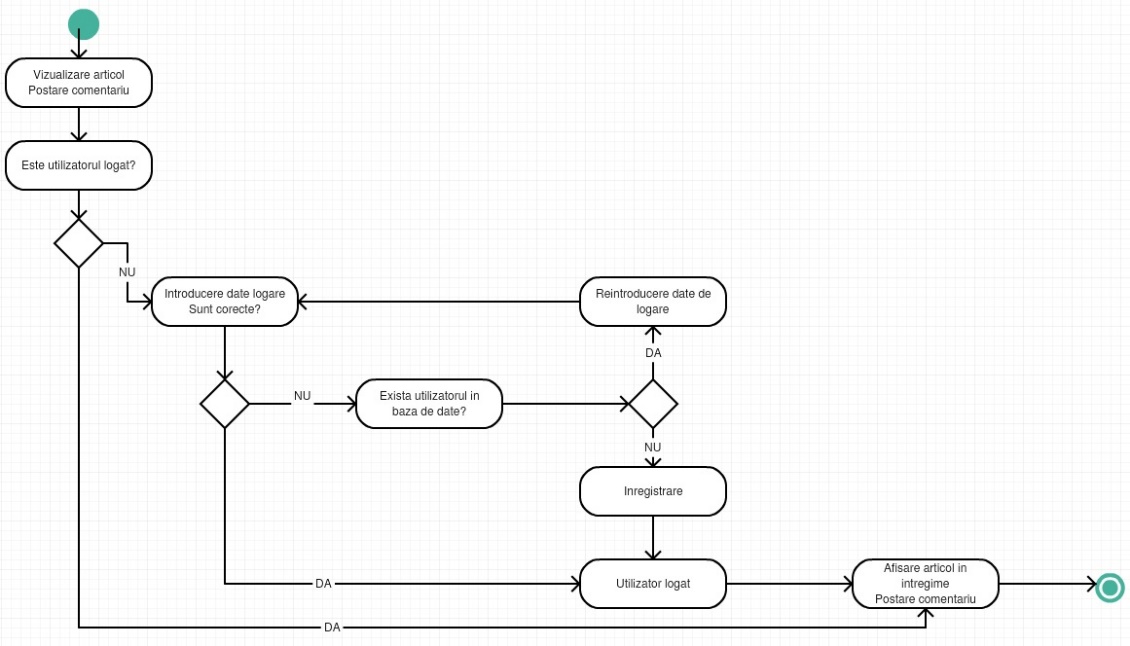


Figura 3: Diagrama de activități interfață – sistem de autentificare/înregistrare.

Următorul pas a fost proiectarea bazei de date conform specificațiilor:

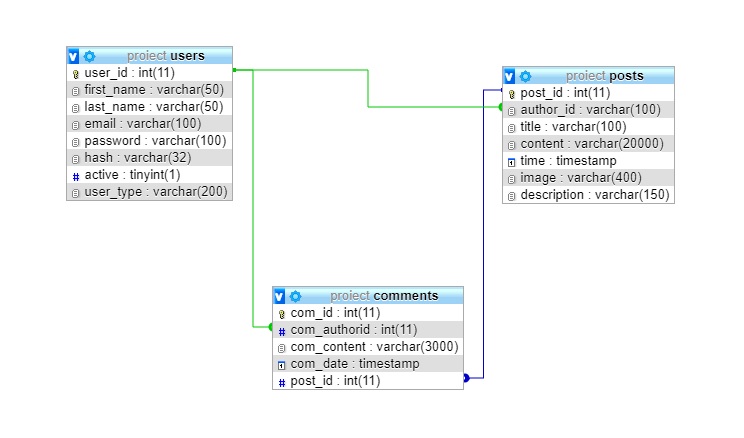


Figura 4: Proiectarea bazei de date

1. **Implementarea aplicației**

**3.1 Tehnologii utilizate**

Pentru realizarea acestui proiect, am utilizat Xamp versiunea 3.2.2 și tehnologii precum PHP 5, MySQL 8, HTML 5, CSS 3, Bootstrap 3.2 și jQuery 1.11.3.

**XAMPP** este un pachet de programe free software, open source și cross-platform web server, care constă în Apache HTTP Server, MySQL database și interpretoare pentru scripturile scrise în limbajele de programare PHP și Perl. XAMPP permite programatorului să dezvolte și să testeze aplicația web pe localhost.

**HTML 5** este a cincea revizuire a standardului HTML (creat în 1990 și standardizat ca HTML4 din 1997) și din octombrie 2011 este în curs de dezvoltare. Obiectivele sale principale au fost acelea de a îmbunătăți limbajul cu un suport pentru cele mai recente apariții multimedia în același timp menținându-l ușor de citit de oameni și bine înțeles de computere și device-uri (browsere web, parsere, etc.). HTML5 își propune să însumeze nu numai HTML4, dar și XHTML1 și DOM2HTML (îndeosebi JavaScript).[[1]](#footnote-1) CSS *(Cascading Style Sheets)* este un standard pentru formatarea elementelor unui document [HTML](https://ro.wikipedia.org/wiki/HTML).  **CSS3** reprezintă un upgrade ce aduce câteva atribute noi și ajută la dezvoltarea noilor concepte in webdesign.

Utilizarea unui limbaj precum **PHP** şi a unei baze de date (MySQL) permite obţinerea unor site-uri web dinamice, care pot fi personalizate şi conţin informaţie în timp real. PHP este un limbaj de tip scripting pe partea de server, proiectat special pentru Web. Într-o pagină HTML poate fi încapsulat cod PHP care va fi executat de fiecare dată când pagina este vizitată. Codul PHP este interpretat pe serverul web şi generează cod HTML (sau un alt output) care va fi văzut de către vizitator.[[2]](#footnote-2)

**MySQL** este un SGBD relaţional recunoscut pentru rapiditate şi robusteţe. Serverul MySQL controlează accesul concurent la date de către mai mulţi utilizatori, care trebuie să fie autorizaţi pentru a obţine accesul respectiv. Prin urmare, serverul MySQL este multiuser şi multi threaded. El utilizează SQL, limbajul standard de gestiune a bazelor de date.

**Bootstrap** reprezintă cel mai popular framework HTML, CSS și Javascript pentru dezvoltarea de proiecte web de tip responsive, adaptabile și pentru ecrane medii (tablete) și mici (telefoane inteligente).

**JQuery** este o bibliotecă Javascript mică, dar care face lucruri precum traversarea și manipularea documentelor HTML, gestionarea evenimentelor, animațiile și multe altele foarte simple, având un API (application program interface) care funcționează pe o multitudine de navigatoare web.

**3.2 Resurse soft necesare**

Una dintre trăsăturile importante atât ale PHP cât şi MySQL este aceea că rulează pe orice sistem de operare important şi multe dintre cele secundare. Majoritatea codului PHP poate fi scris astfel încât să fie portabil între sistemele de operare şi serverele web. Există câteva funcţii PHP care se referă la sistemul de fişiere şi sunt dependente de sistemul de operare.

Odată ce aplicația va fi încărcată pe un server web, aceasta va putea fi accesată de orice utilizator cu un navigator (browser) modern, precum Google Chrome, Opera, Mozilla Firefox, Microsoft Edge etc.

**3.3 Resurse hard necesare**

În prezent, aplicația ocupă aproximativ 4 mb pe disc, dar pe măsură ce utilizatorii vor încărca imagini noi și vor fi publicate noi articole, spațiul necesar de stocare va crește.

În ceea ce privește procesorul, cerințele minime ale aplicației sunt strâns legate de cerințele minime pentru instalarea unui navigator modern. De exemplu, Google Chrome necesită un procesor Intel Pentium 4 sau mai nou.[[3]](#footnote-3)

**3.4 Estimări asupra performanțelor**

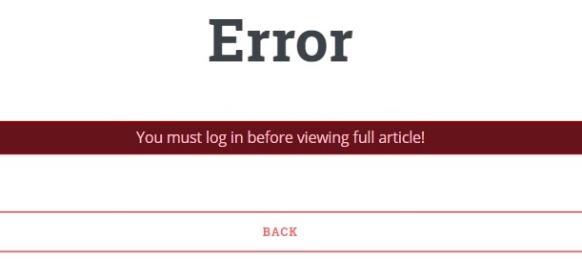
Principalii concurenţi ai PHP sunt Perl, Microsoft ASP.NET, Ruby, Java Server Pages (JSP) şi ColdFusion. Comparativ cu aceste produse, PHP are multe avantaje, inclusiv:

* + performanţă;
  + scalabilitate;
  + interfeţe cu multe SGBD-uri;
  + biblioteci predefinite pentru multe task-uri web comune;
  + cost redus;
  + suport pentru orientarea pe obiecte;
  + portabilitate;
  + flexibilitate în dezvoltare;
  + disponibilitatea codului sursă;
  + disponibilitatea suportului şi a documentaţiei. [[4]](#footnote-4)

În ceea ce privește scalabilitatea PHP, se poate implementa scalarea orizontală cu un număr mare de servere, în mod eficient şi ieftin.

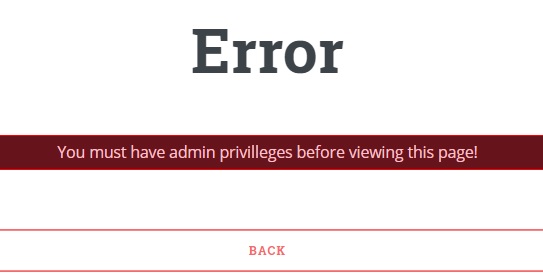
Dintre avantajele MySQL, pot fi enumerate:

* + performanţa ridicată;
  + costul scăzut;
  + uşurinţa configurării şi învăţării;
  + portabilitatea;
  + disponibilitatea codului sursă;
  + disponibilitatea suportului.[[5]](#footnote-5)

1. **Testarea aplicației**

În faza de testare, a fost verificată îndeosebi implementarea cerințelor critice. În ceea ce privește sistemul de autentificare/înregistrare utilizatori, s-a acordat o deosebită atenție testării modului de funcționare a restricționării accesării de conținut neautorizat.

Figura 5: Testarea accesării unui articol de către un utilizator ne-autentificat

****Testele realizate nu sunt exhaustive, dar asigură minimul calității solicitate de beneficiar.

În ceea ce privește interfața, a fost testată intens funcționalitatea acesteia și remediate erorile apărute. Au fost realizate teste pe navigatoare diferite (Google Chrome, Mozilla Firefox, Opera, Microsoft Edge) și pe sisteme de operare diferite (Windows 10, Linux Ubuntu, Android 6).

Figura 6: Testarea accesării panoului de administrator de către un utilizator obișnuit

****

1. **Perspective de dezvoltare ulterioară a aplicației**

Aplicația poate fi dezvoltată ulterior prin adăugarea unor noi categorii de utilizatori. De exemplu, categoria „editori” ar putea prelua din atribuțiunile administratorilor în ceea ce privește publicarea de noi articole. De asemenea, ar putea fi implementată posibilitatea moderării de către editori a comentariilor publicate de utilizatori.

De asemenea, pe măsură ce numărul articolelor publicate pe site va crește, va deveni necesară împărțirea acestora pe diferite categorii, cu editori specializați pentru fiecare categorie.

Figura 7: Interfața este responsive

**Bibliografie**

* Ian Sommerville, Software engineering, Adisson-Wesley, 9th edition, 2011;
* *Programarea interfeţelor web utilizând PHP şi MySQL*, http://193.226.51.37/web/curs;
* http://php.net;
* https://www.w3.org;

1. https://www.w3.org/TR/html5-diff/, consultat în data de 15.04.2018; [↑](#footnote-ref-1)
2. *Programarea interfeţelor web utilizând PHP şi MySQL*, http://193.226.51.37/web/curs/curs1.pdf, consultat la data de 15.04.2018; [↑](#footnote-ref-2)
3. https://support.google.com/chrome/a/answer/7100626?hl=en [↑](#footnote-ref-3)
4. *Programarea interfeţelor web utilizând PHP şi MySQL*, http://193.226.51.37/web/curs/curs1.pdf, consultat la data de 15.04.2018; [↑](#footnote-ref-4)
5. Ibidem. [↑](#footnote-ref-5)