Ministerul Educaţiei, Culturii și Cercetării al Republicii Moldova

Universitatea Tehnică a Moldovei

Departamentul Ingineria Software și Automatică

**Disciplina: Planificarea și estimarea costurilor**

**Lucrare de laborator nr.1**

**Tema: Platfroma digitală pentru anunțuri de**

**vânzare-cumparare a automobilelor**

A efectuat: st.gr.TI-215,

Josan Artiom

A verificat : asist. Univ.

Cojocaru Svetlana

Chișinău 2024

**1. Etapele majore ale proiectului**

1. **Inițializarea proiectului**
   * Identificarea scopului și obiectivelor.
   * Analiza cerințelor generale.
2. **Planificarea proiectului**
   * Crearea unui plan de proiect detaliat.
   * Alocarea resurselor,
   * Definirea specificațiilor tehnice și funcționale.
3. **Dezvoltarea platformei**
   * Crearea arhitecturii platformei,
   * Dezvoltarea interfeței utilizator (UI/UX),
   * Implementarea funcționalităților (e.g., crearea și publicarea anunțurilor).
4. **Testarea platformei**
   * Testarea funcționalităților (e.g., validarea anunțurilor).
   * Identificarea și remedierea erorilor.
   * Realizarea testelor de performanță și securitate.
5. **Lansarea și mentenanța**
   * Lansarea oficială a platformei.
   * Monitorizarea performanței.
   * Asigurarea suportului tehnic și îmbunătățiri periodice.

**Etapa 2: Planificarea proiectului**

**2.1: Crearea WBS și alocarea resurselor**

**Descompunerea etapelor în activități detaliate:**

* Fiecare etapă majoră este împărțită în sarcini și subtasks specifice. De exemplu:
  + Etapa "Dezvoltarea platformei" este divizată în: definirea arhitecturii tehnice, dezvoltarea bazei de date, implementarea interfeței utilizatorului.
  + Subtasks pentru "Definirea arhitecturii tehnice" includ analiza cerințelor arhitecturale și selectarea limbajelor de programare.
  + Subtasks pentru "Dezvoltarea bazei de date" includ proiectarea tabelelor și implementarea relațiilor între ele.

**Identificarea responsabililor pentru fiecare activitate:**

* Fiecare activitate este atribuită unui membru al echipei:
  + Dezvoltator back-end pentru baza de date.
  + Designer UI/UX pentru interfața utilizatorului.
  + Project manager pentru coordonarea generală.

**Alocarea resurselor necesare:**

* Resursele necesare sunt distribuite în funcție de cerințele activităților:
  + **Resurse umane:** dezvoltatori, designeri, tester software.
  + **Hardware:** servere pentru dezvoltare și testare.
  + **Software:** licențe pentru unelte de proiectare și dezvoltare (e.g., Figma, Visual Studio).

**2.2: Realizarea unui plan Gantt pentru planificare**

**Identificarea dependențelor între activități:**

* De exemplu:
  + Designul interfeței (UI/UX) trebuie completat înainte de codarea acesteia.
  + Testarea performanței poate începe doar după ce implementarea modulelor principale este finalizată.

**Stabilirea duratei fiecărei activități și a termenelor-limită:**

* Se alocă timp estimativ pentru fiecare task:
  + Definirea arhitecturii: 10 zile.
  + Dezvoltarea bazei de date: 15 zile.
  + Testarea funcționalităților de bază: 7 zile.
* Termenele-limită sunt setate astfel încât să respecte dependențele dintre activități.

**Monitorizarea progresului proiectului în timp real:**

* Folosirea unui instrument (e.g., Microsoft Project) pentru actualizarea statusului fiecărei activități.
* Progresul este vizualizat ca bare colorate în graficul Gantt.

**2.3: Documentarea specificațiilor tehnice și funcționale**

**Definirea arhitecturii sistemului:**

* Alegerea unei arhitecturi modulare: module separate pentru UI, logică de afaceri și bază de date.
* Utilizarea unui framework precum MVC (Model-View-Controller).

**Cerințe detaliate pentru fiecare funcție a platformei:**

* Publicarea anunțurilor:
  + Formulare cu validări pentru text, imagini și preț.
* Filtre avansate:
  + Opțiuni de sortare după preț, marcă, an de fabricație.
* Mesagerie:
  + Criptare end-to-end pentru mesaje.

**Specificațiile hardware și software necesare:**

* Servere cloud (e.g., AWS) pentru găzduirea platformei.
* Unelte de dezvoltare: framework-uri precum React pentru front-end și Node.js pentru back-end.

**Etapa 3: Dezvoltarea platformei**

**3.1: Definirea arhitecturii tehnice și dezvoltarea bazei de date**

**Alegerea tehnologiilor:**

* Back-end: Node.js sau Python (Django).
* Front-end: React sau Angular.
* Bază de date: MySQL pentru relaționalitate și MongoDB pentru flexibilitate.

**Definirea modelului de date:**

* Tabele:
  + Utilizatori (nume, email, parolă).
  + Anunțuri (titlu, descriere, imagini, preț, ID utilizator).
  + Mesaje (ID expeditor, ID destinatar, text, timestamp).

**Implementarea bazei de date cu relații bine definite:**

* Relații între tabele, de exemplu: fiecare anunț este asociat unui utilizator.

**3.2: Dezvoltarea interfeței pentru utilizatori**

**Crearea unui design intuitiv:**

* Se realizează wireframes în Figma pentru testarea designului înainte de implementare.

**Dezvoltarea interfeței utilizând tehnologii front-end:**

* Componente UI reutilizabile pentru butoane, liste și formulare.
* Stilizare cu CSS modern (e.g., Tailwind CSS).

**Asigurarea compatibilității:**

* Testarea pe browsere principale (Chrome, Firefox, Safari) și dispozitive (desktop, mobil).

**3.3: Implementarea modulelor principale**

**Publicarea și gestionarea anunțurilor:**

* Funcții pentru upload imagini și completare formulare.
* Validare automată pentru câmpuri obligatorii.

**Funcția de căutare avansată:**

* Implementarea filtrelor folosind interogări SQL dinamice.

**Sistemul de mesagerie:**

* Integrarea unui chat în timp real folosind WebSockets.

**Etapa 4: Testarea platformei**

**4.1: Testarea funcționalităților de bază**

**Interfața utilizator:**

* Scenarii de test: navigare între pagini, încărcare anunțuri, utilizarea filtrelor.

**Funcționarea funcționalităților:**

* Validarea corectitudinii operațiilor (e.g., anunțurile sunt salvate corect).

**4.2: Testarea performanței**

**Timpul de răspuns:**

* Testarea timpilor de încărcare pentru diverse operațiuni.

**Stabilitatea:**

* Teste de stres: simularea unui număr mare de utilizatori simultani.

**4.3: Identificarea și remedierea erorilor tehnice**

**Debugging pentru cod:**

* Utilizarea uneltelor precum Debugger sau Chrome DevTools.

**Repetarea testelor:**

* Erorile descoperite sunt remediate, iar testele reluate până la conformitate.

**Etapa 5: Lansarea și mentenanța**

**5.1: Lansarea platformei**

* Publicarea domeniului și verificarea tuturor funcționalităților.

**5.2: Realizarea campaniei de promovare**

* Anunțuri plătite pe Google Ads și Facebook.

**5.3: Monitorizarea și actualizarea funcționalităților**

* Feedback din partea utilizatorilor și îmbunătățiri constante.

**3. Organizarea logică a sarcinilor în WBS**

**Instrument utilizat: Trello**

**Pași pentru organizare:**

1. Crearea unui **Board** pentru proiect (denumit "Dezvoltare Platformă Auto").
2. Crearea de **Liste** pentru fiecare etapă majoră.
3. Adăugarea **Cardurilor** pentru fiecare activitate/sarcină specifică.
4. Alocarea responsabililor pentru fiecare card.
5. Adăugarea de termene-limită și priorități pentru carduri.

**Rezultat final în Trello:**

**Liste:**

1. Inițializare
2. Planificare
3. Dezvoltare
4. Testare
5. Lansare și mentenanță

**Exemple de carduri (sarcini):**

* **În lista "Dezvoltare":**
  + Crearea arhitecturii bazei de date.
  + Dezvoltarea funcției de publicare a anunțurilor.
  + Implementarea notificărilor automate.
* **În lista "Testare":**
  + Testarea interfeței utilizatorului.
  + Verificarea conformității imaginilor încărcate.
  + Realizarea testului de stres al serverului.

**4. Durata estimată pentru realizarea WBS:**

**3 ore academice:**

* 1 oră pentru definirea etapelor și sarcinilor.
* 2 ore pentru organizarea în Trello și verificarea structurii.

Proiectul este planificat în mod logic și detaliat, utilizând metodologia WBS.