**Definirea Cerințelor pentru Aplicația de Gestionare a Pacienților pentru Medicii de Familie**

**Descriere**

**Aplicația este destinată medicilor de familie pentru a facilita gestionarea pacienților, programărilor, consultațiilor, rețetelor și medicamentelor într-un mod eficient și centralizat. Aceasta trebuie să permită medicilor să gestioneze ușor informațiile pacienților și să monitorizeze istoricul medical, diagnosticările, recomandările și medicamentele prescrise.**

**Lista Funcționalităților**

1. **Gestionarea pacienților:**
   * **Adăugarea, modificarea și ștergerea informațiilor despre pacienți.**
   * **Vizualizarea listei pacienților și a detaliilor acestora.**
   * **Căutare avansată după nume, CNP, vârstă, sex, etc.**
2. **Gestionarea programărilor:**
   * **Adăugarea programărilor pentru pacienți.**
   * **Modificarea și anularea programărilor.**
   * **Vizualizarea programărilor pentru o anumită dată și oră.**
   * **Trimiterea notificărilor sau alertelor pentru programări.**
3. **Gestionarea consultațiilor:**
   * **Înregistrarea consultațiilor cu informații despre diagnostic și recomandări.**
   * **Istoric al consultațiilor pentru fiecare pacient.**
   * **Posibilitatea de a genera rapoarte medicale.**
4. **Gestionarea rețetelor:**
   * **Prescrierea rețetelor pentru pacienți.**
   * **Generarea și vizualizarea istoricului rețetelor pentru fiecare pacient.**
5. **Gestionarea medicamentelor:**
   * **Înregistrarea medicamentelor prescrise în cadrul rețetelor.**
   * **Vizualizarea detaliilor despre medicamente, cum ar fi doza, frecvența administrării și durata tratamentului.**
   * **Istoric al medicamentelor prescrise pentru fiecare pacient.**
6. **Raportare:**
   * **Generarea rapoartelor de activitate (consultații, rețete, medicamente, programări) pentru medic.**
   * **Rapoarte cu privire la tratamentele prescrise și utilizarea medicamentelor.**

**Stabilirea Câmpurilor și a Tipurilor de Date**

**Tabelul Pacienti:**

* **pacient\_id**: INT, AUTO\_INCREMENT, PRIMARY KEY
* **nume**: VARCHAR(100), NOT NULL
* **prenume**: VARCHAR(100), NOT NULL
* **data\_nasterii**: DATE, NOT NULL
* **sex**: ENUM('M', 'F'), NOT NULL
* **adresa**: VARCHAR(255)
* **telefon**: VARCHAR(15)
* **email**: VARCHAR(100)
* **data\_inregistrarii**: DATE, DEFAULT CURRENT\_DATE

**Tabelul Medici:**

* **medic\_id**: INT, AUTO\_INCREMENT, PRIMARY KEY
* **nume**: VARCHAR(100), NOT NULL
* **prenume**: VARCHAR(100), NOT NULL
* **specializare**: VARCHAR(100)
* **telefon**: VARCHAR(15)
* **email**: VARCHAR(100)

**Tabelul Consultatii:**

* **consultatie\_id**: INT, AUTO\_INCREMENT, PRIMARY KEY
* **pacient\_id**: INT, FOREIGN KEY (Pacienti.pacient\_id)
* **medic\_id**: INT, FOREIGN KEY (Medici.medic\_id)
* **data\_consultatiei**: DATE, NOT NULL
* **diagnostic**: TEXT
* **recomandari**: TEXT

**Tabelul Retete:**

* **reteta\_id**: INT, AUTO\_INCREMENT, PRIMARY KEY
* **pacient\_id**: INT, FOREIGN KEY (Pacienti.pacient\_id)
* **medic\_id**: INT, FOREIGN KEY (Medici.medic\_id)
* **data\_emiterii**: DATE, NOT NULL
* **detalii\_reteta**: TEXT

**Tabelul Medicamente:**

* **medicament\_id**: INT, AUTO\_INCREMENT, PRIMARY KEY
* **nume\_medicament**: VARCHAR(100), NOT NULL

**Tabelul Reteta\_Medicamente:**

* **reteta\_id INT**
* **medicament\_id INT**
* **doza**: VARCHAR(50), NOT NULL
* **frecventa\_administrare**: VARCHAR(50), NOT NULL
* **durata\_tratament**: INT, COMMENT 'Numărul de zile'

**Tabelul Programari:**

* **programare\_id**: INT, AUTO\_INCREMENT, PRIMARY KEY
* **pacient\_id**: INT, FOREIGN KEY (Pacienti.pacient\_id)
* **medic\_id**: INT, FOREIGN KEY (Medici.medic\_id)
* **data\_programarii**: DATE, NOT NULL
* **ora\_programarii**: TIME, NOT NULL
* **status**: ENUM('Programat', 'Anulat', 'Finalizat'), DEFAULT 'Programat'

**Identificarea Relațiilor dintre Tabele**

1. **Pacienti - Consultatii**: Relație One-to-Many

(Un pacient poate avea mai multe consultații).

1. **Medici - Consultatii**: Relație One-to-Many

(Un medic poate realiza mai multe consultații).

1. **Pacienti - Retete**: Relație One-to-Many

(Un pacient poate primi mai multe rețete).

1. **Medici - Retete**: Relație One-to-Many

(Un medic poate prescrie mai multe rețete).

1. **Retete - Reteta\_Medicamente**: Relație One-to-Many

(O rețetă poate include mai multe medicamente, fiecare cu doză, frecvență și durata tratamentului).

1. **Medicamente - Reteta\_Medicamente**: Relație One-to-Many

**Constrângeri de Integritate**

**Chei Primare:**

* **pacient\_id** pentru tabelul Pacienti.
* **medic\_id** pentru tabelul Medici.
* **consultatie\_id** pentru tabelul Consultatii.
* **reteta\_id** pentru tabelul Retete.
* **medicament\_id** pentru tabelul Medicamente.
* **programare\_id** pentru tabelul Programari.

**Chei Străine:**

* **pacient\_id** și **medic\_id** în tabelele Consultatii, Retete și Programari.
* **reteta\_id** și **medicament\_id** în tabelul intermediar Reteta\_Medicamente.

**Constrângeri de Unicitate:**

* Se poate impune o constrângere pentru ca adresa de email a pacienților și medicilor să fie unică în baza de date.

**Validarea Datelor:**

* Datele de tip **ENUM** (sex, status) asigură că sunt introduse valori predefinite, prevenind erorile de introducere (ex: sexul pacientului să fie doar 'M' sau 'F', statusul programărilor să fie 'Programat', 'Anulat' sau 'Finalizat').

**Interogarii simple**  
  
*SELECT*

*p.programare\_id,*

*p.pacient\_id,*

*p.medic\_id,*

*CONVERT(VARCHAR, p.data\_programarii, 23) AS data\_programarii,*

*CONVERT(VARCHAR, p.ora\_programarii, 8) AS ora\_programarii,*

*p.status,*

*COALESCE(pac.nume, '') AS pacient\_nume,*

*COALESCE(pac.prenume, '') AS pacient\_prenume,*

*COALESCE(med.nume, '') AS medic\_nume,*

*COALESCE(med.prenume, '') AS medic\_prenume*

*FROM Programari p*

*LEFT JOIN Pacienti pac ON p.pacient\_id = pac.pacient\_id*

*LEFT JOIN Medici med ON p.medic\_id = med.medic\_id*

*SELECT*

*r.reteta\_id,*

*r.pacient\_id,*

*r.medic\_id,*

*CONVERT(VARCHAR, r.data\_emiterii, 23) AS data\_emiterii,*

*r.detalii\_reteta,*

*COALESCE(pac.nume, '') AS pacient\_nume,*

*COALESCE(pac.prenume, '') AS pacient\_prenume,*

*COALESCE(med.nume, '') AS medic\_nume,*

*COALESCE(med.prenume, '') AS medic\_prenume*

*FROM Retete r*

*LEFT JOIN Pacienti pac ON r.pacient\_id = pac.pacient\_id*

*LEFT JOIN Medici med ON r.medic\_id = med.medic\_id*

*SELECT COUNT(\*) AS total\_patients*

*FROM Pacienti*

*SELECT COUNT(\*) AS upcoming\_appointments*

*FROM Programari*

*WHERE data\_programarii >= GETDATE()*

*SELECT COUNT(\*) AS active\_doctors*

*FROM Medici*

*SELECT TOP 1*

*m.specializare,*

*COUNT(\*) AS appointment\_count*

*FROM Programari p*

*INNER JOIN Medici m ON p.medic\_id = m.medic\_id*

*GROUP BY m.specializare*

*ORDER BY appointment\_count DESC*

*SELECT TOP 10*

*CONVERT(VARCHAR, data\_programarii, 23) AS date,*

*COUNT(\*) AS count*

*FROM Programari*

*WHERE data\_programarii >= GETDATE()*

*GROUP BY data\_programarii*

*ORDER BY data\_programarii ASC*

*SELECT TOP 5*

*m.specializare,*

*COUNT(\*) AS count*

*FROM Medici m*

*INNER JOIN Programari p ON m.medic\_id = p.medic\_id*

*GROUP BY m.specializare*

**Interogari complexe**

*SELECT TOP 3*

*m.medic\_id,*

*m.nume + ' ' + m.prenume AS doctor\_name,*

*m.specializare,*

*(SELECT COUNT(\*)*

*FROM Programari p*

*WHERE p.medic\_id = m.medic\_id) AS total\_appointments*

*FROM Medici m*

*ORDER BY total\_appointments DESC;*

*SELECT TOP 3*

*p.pacient\_id,*

*p.nume + ' ' + p.prenume AS pacient\_name,*

*p.telefon,*

*p.email,*

*p.adresa,*

*COUNT(\*) AS total\_appointments,*

*(SELECT COUNT(\*)*

*FROM Retete r*

*WHERE r.pacient\_id = p.pacient\_id) AS total\_prescriptions,*

*(SELECT MAX(pr.data\_programarii)*

*FROM Programari pr*

*WHERE pr.pacient\_id = p.pacient\_id) AS last\_appointment\_date*

*FROM Pacienti p*

*INNER JOIN Programari pr ON p.pacient\_id = pr.pacient\_id*

*GROUP BY*

*p.pacient\_id,*

*p.nume,*

*p.prenume,*

*p.telefon,*

*p.email,*

*p.adresa*

*ORDER BY total\_appointments DESC;*

*SELECT TOP 3*

*m.medic\_id,*

*m.nume + ' ' + m.prenume AS doctor\_name,*

*AVG(prescriptions\_per\_patient) AS avg\_prescriptions\_per\_patient*

*FROM (*

*SELECT*

*r.medic\_id,*

*r.pacient\_id,*

*COUNT(r.reteta\_id) AS prescriptions\_per\_patient*

*FROM Retete r*

*GROUP BY r.medic\_id, r.pacient\_id*

*) AS subquery*

*INNER JOIN Medici m ON subquery.medic\_id = m.medic\_id*

*GROUP BY m.medic\_id, m.nume, m.prenume*

*ORDER BY avg\_prescriptions\_per\_patient DESC;*

*SELECT TOP 5*

*m.medic\_id,*

*m.nume + ' ' + m.prenume AS doctor\_name,*

*(SELECT COUNT(\*)*

*FROM Programari p*

*WHERE p.medic\_id = m.medic\_id*

*AND p.status = 'Finalizat') AS completed\_appointments,*

*(SELECT COUNT(\*)*

*FROM Programari p*

*WHERE p.medic\_id = m.medic\_id*

*AND p.status = 'Programat') AS active\_appointments*

*FROM Medici m*

*ORDER BY completed\_appointments DESC;*