**Opgavebeskrivelse: Udvikling af et database- og styringssystem for Hospital Zeta**

**Baggrund:** Hospital Zeta er et moderne hospital, der tilbyder en bred vifte af sundhedstjenester. Hospitalets ledelse har bemærket, at deres nuværende patient- og lægehåndteringssystem er ineffektivt, hvilket resulterer i længere ventetider og dårlig overblik over patientjournaler, aftaler, og lægernes tilgængelighed.

Hospitalet ønsker at digitalisere og optimere deres system til at håndtere patienter, læger, aftaler, medicinske journaler, afdelinger og medicinske diagnoser. Dit opgave er at designe, implementere og administrere et database- og styringssystem for Hospital Zeta, som dækker deres behov og samtidig sikrer datakonsistens og høj ydeevne.

**Krav til systemet:**

Du skal levere et avanceret database-design samt funktionalitet, der opfylder følgende krav:

1. **Databasedesign (Tax 3, Målepind 1)**
   * Udarbejd et E/R-diagram, der beskriver følgende tabeller og deres relationer:
     + **Patienter** (PatientID, Navn, Fødselsdato, Adresse, Telefonnummer, MedicinskHistorik)
     + **Læger** (LaegeID, Navn, Speciale, Telefonnummer, Afdeling)
     + **Aftaler** (AftaleID, PatientID, LaegeID, Dato, Tid, Lokale)
     + **Afdelinger** (AfdelingID, Navn, Beliggenhed)
     + **Journaler** (JournalID, PatientID, LaegeID, Diagnose, Behandling, Dato)
2. **Forespørgsler (Tax 3, Målepind 2, 7)**
   * Implementér SQL-forespørgsler ved brug af SELECT, JOIN og subqueries:
     + Udtræk patienter, der har en given diagnose.
     + Udtræk lægernes antal af konsultationer per afdeling.
     + Udtræk alle aftaler for en specifik læge sorteret efter dato og tid.
3. **Data Manipulation (Tax 3, Målepind 3)**
   * Implementér INSERT, UPDATE og DELETE:
     + Indsæt nye patienter og læger.
     + Opdater patientens medicinske historie.
     + Slet aftaler, der er mere end to år gamle.
4. **Tabeller og Relationer (Tax 3, Målepind 4, 5)**
   * Opret tabeller og relationer mellem patient-, læge- og aftaletabellerne. Dokumentér designet ved hjælp af et E/R-diagram.
   * Implementér 1, mange og 1:1 relationer, hvor det er nødvendigt.
5. **Constraints og Konsistens (Tax 3, Målepind 6, 11)**
   * Sørg for at implementere konsistenskrav og referenceintegritet mellem tabellerne.
   * Definer constraints for at sikre korrekt datavalidering (f.eks. en læge kan ikke have flere aftaler på samme tidspunkt).
6. **Stored Procedures og Triggers (Tax 3, Målepind 12, 13)**
   * Opret en stored procedure til automatisk at opdatere patientens sidste aftaledato.
   * Implementér en trigger, der sikrer, at patientjournaler bliver opdateret, hver gang der oprettes en ny aftale.
7. **Sikkerhed og Performance (Tax 3, Målepind 9, 10, 14, 15, 16)**
   * Opret et View, der kun tillader adgang til følsomme patientoplysninger for autoriseret personale.
   * Implementér beskyttelse mod SQL Injection i alle forespørgsler.
   * Opret indekser for at optimere ydeevnen i databasen.
   * Anvend performance-måling på store datamængder for at evaluere og optimere systemet.
8. **Databaseadministration (Tax 2, Målepind 8)**
   * Opret og implementér en backup-strategi for at sikre dataintegritet og genopretning i tilfælde af fejl.
9. **ORM og Domænemodel (Tax 3, Målepind 18, 19, 20)**
   * Opret en domain-model, og lad ORM generere databasen og forespørgslerne.
   * Brug ORM til at generere forespørgsler og synkronisere med en eksisterende hospitalsdatabase.
10. **Yderligere overvejelser (Tax 3, Målepind 21, 22, 23, 24)**
    * Diskutér fordele og ulemper ved at vælge en relationel database, ORM eller DocumentDatabase, baseret på hospitalets behov for performance og skalerbarhed.
    * Implementér sikkerhedsforanstaltninger som kryptering af data og password-beskyttelse.

**Aflevering:**

* Dokumentation af E/R-diagram, databasedesign og implementeringsvalg.
* SQL-scripts til oprettelse af tabeller, forespørgsler, views, stored procedures og triggers.
* Resultater fra performance-målinger og sikkerhedsovervejelser.

**Evaluering:**

Din løsning vil blive evalueret ud fra opfyldelsen af ovenstående krav, med særlig fokus på korrekt brug af SQL, sikkerhedsaspekter, datakonsistens, performance-optimering og dokumentation.

God fornøjelse med opgaven!