

Hasta Takip ve Yönetim Sistemi Projesi

Kocaeli Üniversitesi

1. Cihangir Emre Er
Bilgisayar Mühendisliği
Kocaeli Üniversitesi
Kocaeli, Türkiye
220202036@kocaeli.edu.tr

2. Berker Yiğit
Bilgisayar Mühendisliği
Kocaeli Üniversitesi
Kocaeli, Türkiye
220202046@kocaeli.edu.tr

I. Giriş (HEADING I)

Hastane yönetim sistemleri, sağlık hizmetlerinin etkin bir şekilde yönetilmesini sağlayan önemli araçlardır. Bu proje, modern teknolojiler kullanılarak hastane yönetim süreçlerini dijitalleştirmeyi hedeflemektedir. Veri tabanı tasarımı, nesne yönelimli programlama, güvenlik önlemleri ve arayüz geliştirmeleri gibi çeşitli bileşenlerin bir araya getirilmesiyle birlikte, kullanıcıların sağlıkla ilgili işlemleri kolayca gerçekleştirebilecekleri bir platform oluşturulmuştur.

Özet

Bu rapor, bir hastane yönetim sistemi olarak tasarlanmış olan "Hasta Takip ve Yönetim Sistemi" projesini tanıtmaktadır. Projenin amacı, hastaların kayıt oluşturabileceği, doktorlarla randevu alabileceği, tıbbi raporları saklayabileceği ve genel olarak sağlıkla ilgili işlemleri yönetebileceği bir web uygulaması geliştirmektir. Rapor, projenin amaçlarını, tasarımını, geliştirme sürecini ve sonuçlarını detaylı olarak açıklamaktadır.

II. YÖNTEM

Proje geliştirme süreci aşağıdaki adımları içermektedir:

Veri Tabanı Tasarımı: İlk olarak, projenin gereksinimlerine göre veri tabanı tasarlanmıştır. Temel tablolar belirlenmiş ve ilişkiler kurulmuştur. Veri tabanı normalizasyon kurallarına uygun olarak oluşturulmuştur.

Nesne Yönelimli Programlama: Her bileşen için uygun sınıflar oluşturulmuş ve gerekli metotlar tanımlanmıştır. Trigger fonksiyonları yazılarak veri tabanındaki güncellemelerin yönetimi sağlanmıştır.

Güvenlik Önlemleri: Hassas verilerin güvenliği için HTTPS protokolleri ve şifreleme yöntemleri kullanılmıştır.

Arayüz Geliştirmeleri: Kullanıcıların kolayca kullanabilecekleri bir arayüz tasarlanmıştır. AJAX çağrıları kullanılarak dinamik bileşenler eklenmiş ve bildirim sistemi

kurulmuştur..

```

<!-- lang="tr" -->
<!-- head -->
<!-- body -->
<!-- header -->
<div class="nav navbar-expand-lg navbar-light bg-light">
  <a class="navbar-brand" href="#">Kocaeli Hospital Hasta Arayüzü</a>
  <button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent" aria-controls="navbarSupportedContent" data-expanded="true">
    <span class="navbar-toggler-icon"></span>
  </button>
  <div class="collapse navbar-collapse" id="navbarSupportedContent">
    <ul class="navbar-nav mr-auto">
      <li class="nav-item">
        <a class="nav-link" href="http://127.0.0.1:8080/ Randevularin/Randevularin"></a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="http://127.0.0.1:8080/profile/Profile"></a>
      </li>
      <li class="nav-item">
        <a class="nav-link" href="http://127.0.0.1:8080/searchDoctor/Doktorlar"></a>
      </li>
    </ul>
  </div>
</div>
<!-- footer -->
<div class="container mt-4">
  <div class="text-center">Hesapla</div>
</div>

```

```

def login_view(request):
    if request.method == 'POST':
        user_type = request.POST.get('user_type')
        username = request.POST.get('username')
        password = request.POST.get('password')

        conn = sqlite3.connect('hospital.db')
        cursor = conn.cursor()

        # Determine the appropriate SQL query based on the user type
        if user_type == 'patient':
            cursor.execute('SELECT hastaSifre FROM Hastalar WHERE HastaID=?', [username])
        elif user_type == 'doctor':
            cursor.execute('SELECT doktorSifre FROM Doktorlar WHERE DoktorID=?', [username])
        elif user_type == 'nurse':
            cursor.execute('SELECT yoneticisifre FROM Yoneticiler WHERE YoneticiiD=?', [username])
        else:
            messages.error(request, 'Geçersiz kullanıcı türü.')
            return render(request, "hospital/homePage.html")

        # Fetch the result from the executed query
        row = cursor.fetchone()

        # Check if a result was found and if the password matches
        if row is not None and row[0] == password:
            # Redirect to the appropriate user interface
            if user_type == 'patient':
                return render(request, "hospital/hastaArayuz.html")

```


III. DENEYSEL SONUÇLAR

Projenin geliştirme süreci başarıyla tamamlanmıştır. Arayüz üzerinden yapılan işlemler veri tabanında doğru bir şekilde yansıtılmakta ve kullanıcılara geri bildirimler verilmektedir. Güvenlik önlemleri sayesinde kullanıcı verileri korunmakta ve gizlilik sağlanmaktadır.,

```
<html lang="tr">
<body>
<header>
</nav>
</header>
<main class="container mt-4">
<div class="text-center">Hoşgeldiniz!</div>
</-- Ana sayfa içeriği buraya gelecek -->
</main>
<footer class="footer mt-auto py-3 bg-light">
<div class="container text-center">
<!-- Footer içeriği buraya gelecek -->
</div>
</footer>

<!-- Bootstrap JS dosyaları (jQuery ve Popper.js içerir) -->
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.5.4/dist/umd/popper.min.js"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
</body>
</html>
```

```
<!-- Bootstrap JS dosyaları (jQuery ve Popper.js içerir) -->
<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.5.4/dist/umd/popper.min.js"></script>
<script src="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>
</body>
</html>
```

IV. SONUÇ

"Hasta Takip ve Yönetim Sistemi" projesi, hastane yönetim süreçlerini dijitalleştirme ve kullanıcı deneyimini geliştirme amacıyla başarıyla tamamlanmıştır. Projede kullanılan yöntemler ve teknolojiler, benzer uygulamaların geliştirilmesinde de kullanılabilir.

V. Psueodo Kod

Fonksiyon: login_view(request)

Amaç: Kullanıcının giriş yapmasını sağlar ve kullanıcı türüne göre doğru arayüze yönlendirir.

Parametreler:

request: İstek nesnesi.

İşlemler:

Eğer istek yöntemi POST ise:

Kullanıcı türünü, kullanıcı adını ve şifreyi al.

Veritabanına bağlan.

Kullanıcı türüne göre uygun SQL sorgusunu oluştur ve kullanıcı adıyla birlikte çalıştır.

Sorgu sonucunu al.

Eğer sonuç varsa ve şifre eşleşiyorsa:

Kullanıcı türüne göre uygun arayüzü göster.

Eğer sonuç yoksa veya şifre eşleşmiyorsa:

Hata mesajı göster ve ana sayfaya yönlendir.

Eğer istek yöntemi GET ise:

Ana sayfayı göster.

Fonksiyon: index(request)

Amaç: Ana sayfayı gösterir.

Parametreler:

request: İstek nesnesi.

İşlemler:

Ana sayfayı göster.

Fonksiyon: Doctors(request)

Amaç: Doktorlar sayfasını gösterir.

Parametreler:

request: İstek nesnesi.

İşlemler:

Doktorlar sayfasını göster.

Fonksiyon: SearchDoctor(request)

Amaç: Verilen sorguya göre doktor arar ve sonuçları gösterir.

Parametreler:

request: İstek nesnesi.

İşlemler:

Eğer istek yöntemi POST ise:

Arama sorgusunu al.

Veritabanına bağlan.

Doktorları ad veya soyada göre ara.

Sonuçları al.

Sonuçları göster.

Eğer istek yöntemi GET ise:

Boş arama formunu göster.

Fonksiyon: SearchPatient(request)

Amaç: Verilen sorguya göre hastaları arar ve sonuçları gösterir.

Parametreler:

request: İstek nesnesi.

İşlemler:

Eğer istek yöntemi POST ise:

Arama sorgusunu al.

Veritabanına bağlan.

Hastaları ad veya soyada göre ara.

Sonuçları al.
Sonuçları göster.
Eğer istek yöntemi GET ise:
Boş arama formunu göster.
Fonksiyon: Manager(request)

Amaç: Yönetici panelini yönetir.

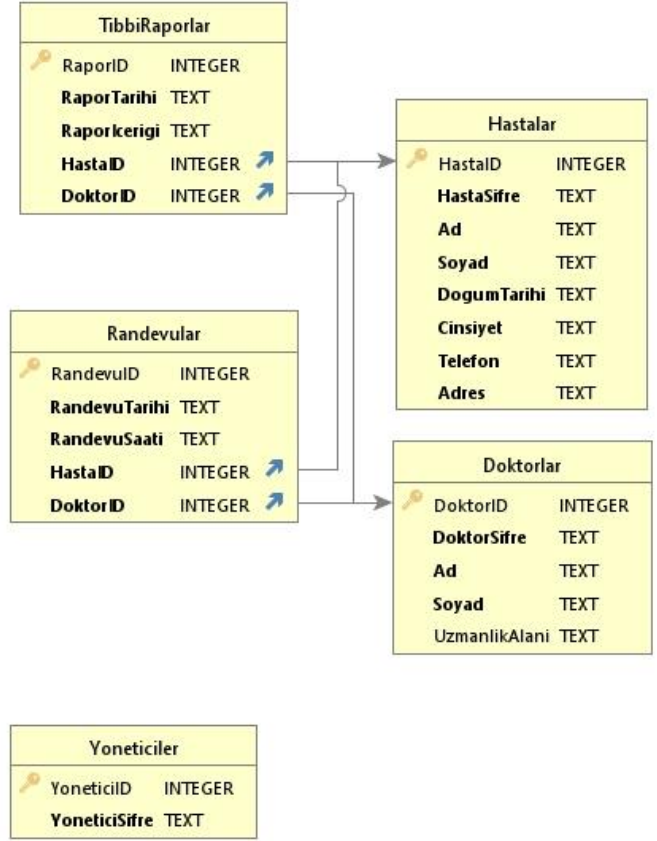
Parametreler:

request: İstek nesnesi.

İşlemler:

Eğer istek yöntemi POST ise:
Eğer 'sil_doktor' butonuna tıklanmışsa:
Silinecek doktorun kimliğini al.
Veritabanından doktoru sil.
Başarılı silme mesajını göster.
Eğer 'sil_hasta' butonuna tıklanmışsa:
Silinecek hastanın kimliğini al.
Veritabanından hastayı sil.
Başarılı silme mesajını göster.
Eğer 'DoktorID' formu doluysa:
Yeni doktor bilgilerini al.
Veritabanına yeni doktoru ekle.
Eğer 'HastaID' formu doluysa:
Yeni hasta bilgilerini al.
Veritabanına yeni hastayı ekle.
Veritabanından mevcut doktor ve hasta bilgilerini al.
Doktor ve hasta bilgilerini göster.

VI. ER DİAGRAMI



KAYNAKLAR

- [1] <https://www.dbvis.com>
- [2] <https://www.youtube.com/watch?v=nwZLJiZ7-oQ>
- [3] https://home.bigid.com/demo-governance?utm_campaign=EUROPE+-+Data+Governance&utm_source=google&utm_medium=cpc&utm_term=data%20mapping%20tool&hsa_ver=3&hsa_cam=16774635055&hsa_grp=138179567227&hsa_net=adwords&hsa_ad=591155408608&hsa_kw=data%20mapping%20tool&hsa_mt=b&hsa_acc=9924524058&hsa_src=g&hsa_tgt=kwd-10541561&gad_source=1&gclid=Cj0KCQjwxqayBhDFARIsAANWRnRgHswz2wSUMCjhRsqPBwvfjwmJwHUvsmAlneTBHqKD-jJ4TYrq5QaAvPhEALw_wcB
- [4] * <https://www.bilisimkonulari.com/proteus-isis-ve-ares-kullanimi-resimli-anlatim.html>
- [5] * <https://www.circuitstoday.com/proteus-software-introduction>
- [6]
- [7] * <https://leetcode.com>

```
<html lang="en">
<body>
  <div class="container">
    <div class="row mt-4">
      <div class="col">
        <table>
          <tr>
            <td>
              <input type="text" id="DoktorID" name="DoktorID" class="form-control">
            </td>
          </tr>
        </table>
      </div>
      <div class="col">
        <input type="password" id="DoktorSifre" name="DoktorSifre" class="form-control">
      </div>
      <div class="col">
        <input type="text" id="Ad" name="Ad" class="form-control">
      </div>
      <div class="col">
        <input type="text" id="Soyad" name="Soyad" class="form-control">
      </div>
      <div class="col">
        <input type="text" id="UzmanlikAlani" name="UzmanlikAlani" class="form-control">
      </div>
      <div class="col">
        <button type="submit" class="btn btn-primary">Doktor Ekle</button>
      </div>
    </div>
  </div>
</body>
</html>
```