

YazLab 2 Proje 3

Buğrahan TOPAL 190201050, Bahadır IŞIK 190201049

1. ÖZET

JavaScript ,nodejs , React kullanarak projeyi yaptık. Projemiz kısaca bir arama sistemi olup burada Neo4j den çektiğimiz verileri gösterdiğimiz,arayabildiğimiz bir yapıdır.

2. GİRİŞ

Bu projede ilk amacımız Javascript kullanarak bir localhost oluşturmaktı. Daha sonra database'i javascripte entegre etmek oldu. Entegre edilen database ile araştırmacı , yayın gibi verileri neo4j den çektik.

Çektiğimiz verileri Kullanıcı panelinde arama ve gösterme işlemini yaptık.Admin panelinde ise database'e ekleme işlemini yaptık.Ayrıca kullanıcı panelinde Neo4j node gösterimi yaptık.

3. YÖNTEM

3.A)Neo4j

3.B)React

3.C)Javascript

3.D)React-Dom

3.F)React-Resize-Aware

3.G)Neovis

4. DENEYSEL SONUÇLAR

4.A) Ana Giriş Ekranı :

KOU Arama Sistemi

Login

Email:

Password:

4.B) Kullanıcı Paneli Giriş

KOU Arama Sistemi

Kullanici Paneli

4.C) Admin Paneli Giriş

KOU Arama Sistemi

Admin Paneli

4.D) Kullanıcı Arama Ekranı



- * Kodda bu kullanıcıların olup olmadığını sorguladık.
- * Varsa kullanıcıyı dashboardına yönlendirdik.
- * Yoksa hata verdirdik ve giriş yapamadı.
- * Ayrıca kullanıcı doğru girişi yaptığında ekranına yolladık
- * Giriş yapan kullanıcıyı kullanıcı panelinden ekranına yolladık
- * User'a giden kullanıcı Router üzerinden kullanıcı' ya yönlendi .
- * Burada arama yapılabilir
- * Arama yaptığımız kişinin id sini alıyoruz.
- * ID si ile onun sayfasına yönlendirme yapıyoruz.
- * Burada yazarın çalıştığı kişiler gözüküyor.
- * Ayrıca burada yazarın üzerinde çalıştığı yayınlar gözüküyor.
- * Burada birlikte çalıştığı kişiler tıklanabilir.
- * Kişiyi tıkladığımızda arama dizisine gider.
- * Bu dizide tıkladığımız kişinin id si alınır.

5.AKIŞ DİYAGRAMI

- * Neo4j database'ine veri ekledik.
- * Neo4j içinde ilişkiler oluşturduk
- * Gerekli tabloları oluşturduk.
- * localhostumuza ulaşmak için 3000 de bir port kurduk.
- * localhostumuz olmuş oldu.
- * Projemize neo4j bağlantısını kurduk bu sayede bize gönderilen verileri neo4j üzerinden karşılaştıracamız.
- * Projemizin kullanıcı girişini yapacağı login formun html ve css gibi yerlerini hazırladık.
- * Localhost 3000 üzerine / noktasına bir get gönderdik.
- * Bu sayede / adresinde login form görünür oldu.
- * / adresi ile gelen adresler için bir router oluşturduk.
- * Router bizi gerekli sayfalara yönlendirdi.
- * LoginForm içindeki LoginForm fonksiyonu sayesinde login formumuz görünür hale geldi.
- * Login formun içine kullanıcı adı ve şifre isteyen bölümü koyduk.
- * Ve bu formu post olarak atadık.
- * Router üzerinden / post olan adresi kurduk bu adres login user fonksiyonunu çağırdı
- * Kullanıcının en üstüne neo4j verileri için kullandığımız databasei ekledik.
- * Bu sayede kolay bir erişim sağlamış olduk.
- * Formdan gelen username ve password verilerini gerekli yerlerde tuttuk.
- * ID si alınan kişi yine aynı şekilde kendi sayfasına yönlendirilir.
- * Burada yine onun çalıştığı kişiler ve yayınları gözükür.
- * Aynı şekilde onun çalıştığı kişiler tıklanabilir. Bu kişilere tıkladığımızda yine aynı şekilde onun sayfasına gider.
- * Arama sayfasında neo4j ' e bağlandık.
- * Neo4j node gösterimi yaptık.
- * Admin sayfası da aynı şekilde admin paneline giden yer vardır.
- * Admin sayfasında yazar yayın gibi eklemeler yapılır.

6.SONUÇ

Bu proje sayesinde Javascript ile site yazmayı öğrendik. Neo4j kullanmamız farklı database yapılarını kullanmayı öğrendik . Neovis sayesinde node gösterimi yapmayı öğrendik. React ile çalışmayı öğrendik.

7. KAYNAKÇA

- * <https://github.com/jackdbd/react-neovis-example>
- * <https://tr.reactjs.org/docs/getting-started.html>
- * <https://neo4j.com/docs/cypher-manual/current/clauses/merge/>
- * <https://neo4j.com/docs/cypher-manual/current/clauses/>