

# Bulut Bilişim ile Dosya Depolama Uygulaması Projesi

Proje linki:

[https://drive.google.com/file/d/1j1kLlB\\_FubLv0ToRLKEkcGYoH6g5\\_0Pxu/vie w?usp=sharing](https://drive.google.com/file/d/1j1kLlB_FubLv0ToRLKEkcGYoH6g5_0Pxu/vie w?usp=sharing)

Aydın Kasımoğlu  
Kocaeli Üniversitesi Teknoloji Fak.  
Bilişim Sistemleri Mühendisliği  
211307011

**Özet—**Bulut Bilişim ile Dosya Depolama Uygulaması Projesi, bir web uygulamasının geliştirilmesinde bulut bilişim altyapısı ve sanallaştırma teknolojilerinin entegrasyonunu araştırmaktadır. Bu proje, kullanıcı kimlik doğrulaması için AWS Cognito, dosya depolama için AWS S3 ve ön uç geliştirme için NextJS kullanarak dosyaları bulutta güvenli bir şekilde depolamaya odaklanmaktadır. Rapor, uygulamanın teorik temellerini, metodolojilerini ve mimari tasarımını detaylandırmaktadır. Bu proje, bulut bilişim ve sanallaştırmayı inceleyerek, modern yazılım geliştirmenin gelişen ortamına katkıda bulunmayı ve bu teknolojilerin uyumlu ve ölçeklenebilir bir web uygulamasında pratik olarak uygulanmasını göstermeyi amaçlamaktadır. Rapor, projenin kavramsallaştırılmasından bulut bilişim ve sanallaştırma teknolojileri arasındaki simbiyotik ilişkinin vurgulanmasına kadar kapsamlı bir yolculuğu kapsamaktadır.

**Anahtar kelimeler—**Bulut Bilişim, Sanallaştırma, Kimlik Doğrulama, Dosya Depolama, Bulut Tabanlı Uygulamalar

## I. GİRİŞ

Bulut Bilişim Sanallaştırma Projesi, modern bilişimin sürekli gelişen ortamına yanıt olarak, bulut bilişim altyapısı ve sanallaştırma teknolojilerinin yeteneklerini bir araya getirmeye çalışmaktadır. Bu proje, bu iki önemli alan arasındaki sinerjiden yararlanan sofistike bir web uygulamasının geliştirilmesine odaklanmaktadır.

Bulut bilişim, ölçeklenebilir ve isteğe bağlı kaynaklarıyla, farklı hesaplama ihtiyaçları olan uygulamaları barındırmak için ideal bir temel sağlar. Aynı zamanda, sanallaştırma teknolojileri bu kaynakları soyutlamak ve yönetmek, esnekliği artırmak ve kaynak kullanımını optimize etmek için etkili bir yol sunar.

Projemizde bulut bilişim ve sanallaştırmanın bir araya getirilmesi, sağlam bir web uygulamasının oluşturulması gibi önemli bir ihtiyacı karşılamaya yöneliktir. Dosyaları bulutta güvenli bir şekilde saklamak için tasarlanan bu uygulama, kullanıcı kimlik doğrulaması için AWS Cognito, verimli dosya depolama için AWS S3 ve dinamik ön uç geliştirme için NextJS'in sorunsuz bir entegrasyonunu sergiliyor.

## II. LİTERATÜR TARAMASI

### A. Bulut Bilişim

Bulut bilişim hem bir platformu hem de uygulama türünü tanımlamak için kullanılan bir terimdir. Bir bulut bilişim platformu, sunucuları dinamik olarak hazırlar, yapılandırır, yeniden yapılandırır ve gerektiğinde sunucuları kaldırır. Buluttaki sunucular fiziksel makineler veya sanal makineler

olabilir. Gelişmiş bulutlar tipik olarak depolama alanı ağları (SAN'lar), ağ ekipmanları, güvenlik duvarı ve diğer güvenlik cihazları gibi diğer bilgi işlem kaynaklarını içerir [1].

Bulut bilişim ayrıca İnternet üzerinden erişilebilecek şekilde genişletilmiş uygulamaları da tanımlar. Bu bulut uygulamaları, Web uygulamalarını ve Web hizmetlerini barındıran büyük veri merkezleri ve güçlü sunucular kullanır. Uygun bir İnternet bağlantısı ve standart bir tarayıcıya sahip olan herkes bir bulut uygulamasına erişebilir [1].

NIST (National Institute of Standards and Technology) bulut bilişimin temel özelliklerini şu şekilde açıklıyor:

- **Talep üzerine self-servis:** Bir tüketici, sunucu süresi ve ağ depolama alanı gibi bilgi işlem yeteneklerini, her bir hizmet sağlayıcıyla insan etkileşimi gerektirmeden otomatik olarak gerektiği şekilde tek taraflı olarak sağlayabilir.
- **Geniş ağ erişimi:** Olanaklar ağ üzerinden kullanılabilir ve heterojen ince veya kalın istemci platformları (örn. cep telefonları, tabletler, dizüstü bilgisayarlar ve iş istasyonları) tarafından kullanımı teşvik eden standart mekanizmalar aracılığıyla erişilebilir.
- **Kaynak havuzu:** Sağlayıcının bilgi işlem kaynakları, çok kiracılı bir model kullanılarak birden fazla tüketiciye hizmet vermek üzere bir havuzda toplanır ve farklı fiziksel ve sanal kaynaklar tüketici talebine göre dinamik olarak atanır ve yeniden tahsis edilir. Müşterinin genellikle sağlanan kaynakların tam konumu üzerinde hiçbir kontrolü veya bilgisi olmadığı, ancak daha yüksek bir soyutlama düzeyinde (örneğin, ülke, eyalet veya veri merkezi) konumu belirleyebildiği için konum bağımsızlığı duygusu vardır. Kaynaklara örnek olarak şunlar verilebilir: Depolama, işleme, bellek ve ağ bant genişliği.
- **Hızlı esneklik:** Yetenekler, taleple orantılı olarak hızla dışa ve içe doğru ölçeklendirmek için bazı durumlarda otomatik olarak elastik bir şekilde sağlanabilir ve serbest bırakılabilir. Tüketici için, tedarik için mevcut kabiliyetler genellikle sınırsız gibi görünür ve herhangi bir zamanda herhangi bir miktarda tahsis edilebilir.
- **Ölçülen hizmet:** Bulut sistemleri, hizmet türüne (örneğin depolama, işleme, bant genişliği ve aktif kullanıcı hesapları) uygun bir soyutlama düzeyinde bir ölçüm özelliğinden yararlanarak kaynak kullanımını

otomatik olarak kontrol eder ve optimize eder. Kaynak kullanımı izlenebilir, kontrol edilebilir ve raporlanabilir, böylece kullanılan hizmetin hem sağlayıcısı hem de tüketicisi için şeffaflık sağlanır.

### B. Sanallaştırma

Bulut bilişim, kullanıcıların büyük miktarda veriye, bilgiye ve diğer çeşitli bilgi işlem kaynaklarına, müşterinin bu kaynakları kullandığı ve buna göre ödeme yaptığı bir şekilde erişmesine olanak tanıyan bir teknoloji olarak söylenebilir. Sanal makineden bahsedecek olursak, bulut bilişimin verimliliğini artırmaya yardımcı olan bir makinedir. Sanallaştırma sayesinde, aynı sunucu üzerinde aynı anda birden fazla işletim sistemi ve uygulama üzerinde çalışmak mümkündür, dolayısıyla sanallaştırma donanımın faydasını ve esnekliğini artırır [2].

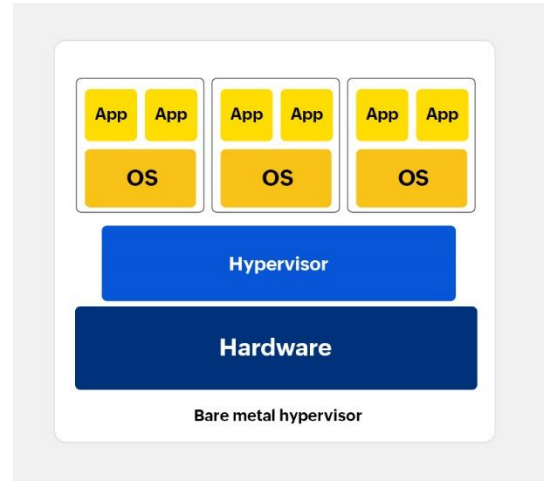
Sanallaştırma, en basit ve kolay kelimelerle bilgisayar kaynaklarının soyutlanmasıdır. Sanallaştırma, bulut bilişim içerisinde işletim sisteminin üzerinde çalıştığı donanımdan ayrılmasını sağlayan bir bileşendir. Sanallaştırma, tek bir fiziksel kaynağın birden fazla sanal kaynak olarak hizmet vermesine izin verme ve hatta birden fazla fiziksel kaynağın tek bir sanal kaynak olarak işlev görmesini sağlama yeteneğine sahiptir. Sanallaştırma, BT kuruluşlarının çalışmalarında her yönden büyük bir değişiklik getirmiştir [2].

### C. Bulut Bilişimde Sanallaştırmanın Rolü

Sanallaştırma ve çok görevli işletim sistemleri benzer çalışma yeteneklerine sahiptir. Bir dizi sanal sunucunun tek bir fiziksel makinede merkezileştirilmesine olanak tanır. Bir kuruluşun belirli bir görev için iki veya daha fazla sunucuyu çalıştırması gerekiyorsa, birinin arızalanması durumunda, hiçbirisi tam kaynak kullanımına yakın olmasa bile. Sanallaştırma burada yardımcı olabilir, çünkü bir fiziksel bilgisayardan diğerine geçiş nispeten kolaydır. Dolayısıyla bir başka büyük avantaj da taşıma işlemidir. Sanallaştırma enerji tasarrufuna yardımcı olabilir, maliyeti ve sunucu donanım ayak izini azaltabilir ve böylece şirketlerin karlarını en üst düzeye çıkarmalarına yardımcı olabilir [2].

Bilgi işlem sanallaştırmasında, donanım ve sanal makine (üzerinde bir işletim sisteminin çalıştığı) arasında bir sanallaştırma katmanı bulunur. Sanallaştırma katmanı hipervizör olarak da bilinir. Hipervizör, sanal makinelere standartlaştırılmış donanım kaynakları (örneğin: CPU, Bellek, Ağ, vb.) sağlar [2].

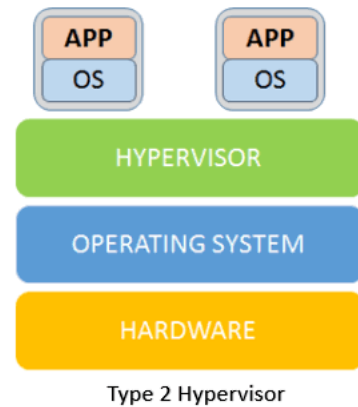
"Tip 1" (veya "çıplak metal") ve "Tip 2" (veya "barındırılan") olarak adlandırılan iki ana hipervizör tipi vardır. Tip 1 hipervizör hafif bir işletim sistemi gibi davranır ve doğrudan ana bilgisayarın donanımı üzerinde çalışırken, tip 2 hipervizör diğer bilgisayar programları gibi bir işletim sistemi üzerinde bir yazılım katmanı olarak çalışır.



Şekil 1. Çıplak metal hipervizör [3].

En yaygın kullanılan hipervizör türü, sanallaştırma yazılımının doğrudan işletim sisteminin normalde kurulu olduğu donanım üzerine kurulduğu tip 1 veya çıplak metal hipervizördür. Çıplak-metal hipervizörler saldırıya eğilimli işletim sisteminden izole edildikleri için son derece güvenlidirler. Buna ek olarak, genellikle barındırılan hipervizörlerden daha iyi ve daha verimli performans gösterirler. Bu nedenlerden dolayı, çoğu kurumsal şirket veri merkezi bilgi işlem ihtiyaçları için bare-metal hipervizörleri seçmektedir.

Çıplak metal hipervizörler doğrudan bilgi işlem donanımı üzerinde çalışırken, barındırılan hipervizörler ana makinenin işletim sisteminin (OS) üzerinde çalışır. Barındırılan hipervizörler işletim sistemi içinde çalışsa da, hipervizörün üzerine ek (ve farklı) işletim sistemleri kurulabilir. Barındırılan hipervizörlerin dezavantajı, gecikme süresinin çıplak metal hipervizörlerden daha yüksek olmasıdır. Bunun nedeni, donanım ve hipervizör arasındaki iletişimin işletim sisteminin ekstra katmanından geçmesi gerekesidir. Barındırılan hipervizörler bazen istemci hipervizörler olarak da bilinir çünkü çoğunlukla son kullanıcılar ve yazılım testleri için kullanılırlar ve yüksek gecikme daha az endişe yaratır.



Şekil 2. Barındırılan hipervizör [4].

## III. MATERYALLER

### 1) AWS Cognito

Amazon Cognito, projede sağlam kullanıcı kimlik doğrulaması için köşe taşı olarak hizmet vermektedir. AWS Cognito, sorunsuz ve güvenli kullanıcı kaydı, oturum açma ve erişim kontrolü işlevlerini kolaylaştırır. Proje, Cognito'nun kapsamlı kimlik hizmetlerinden yararlanarak, kullanıcı verilerinin bütünlüğünü ve gizliliğini korumak için çok önemli olan güvenilir ve ölçeklenebilir bir kimlik doğrulama süreci sağlıyor.

## 2) AWS S3

Amazon Simple Storage Service (S3) projenin dosya depolama altyapısında önemli bir rol oynamaktadır. Son derece ölçeklenebilir ve dayanıklı bir nesne depolama hizmeti olarak işlev gören AWS S3, dosyaları depolamak ve almak için güvenli bir depo sağlar. Sürüm oluşturma ve erişim kontrolü gibi güçlü özellikleri, projenin bulutta kullanıcı tarafından yüklenen içeriği yönetme ve koruma becerisine katkıda bulunur.

## 3) NextJS

Web uygulamasının ön uç geliştirmesi, dinamik ve sunucu tarafından işlenen web uygulamaları oluşturmada verimliliğiyle tanınan React tabanlı bir çerçeve olan NextJS aracılığıyla yönetilmektedir. NextJS, performansı optimize etmeye odaklanarak duyarlı ve görsel olarak çekici kullanıcı arayüzlerinin oluşturulmasını kolaylaştırır. Sunucu tarafı işleme yetenekleri daha hızlı sayfa yüklemelerine katkıda bulunarak genel kullanıcı deneyimini geliştirir.

## IV. METOTLAR

AWS Cognito ile kullanıcı kimlik doğrulaması için ilk önce AWS'nin sağlamış olduğu Cognito konsolundan bir kullanıcı havuzu oluşturdum. Kaydolan kullanıcılar bu havuzda toplanacaktı. Kullanıcı havuzunu oluşturduktan sonra bu havuza erişmek için gerekli bilgileri kaydettim.

Uygulamanın basitçe ana sayfasını, giriş ve kaydolma sayfasını NextJS sayesinde tasarladım ve bu sayfalara, AWS'nin sağladığı bir Javascript çerçevesiyle, Cognito'yu entegre ettim.

Kullanıcı oturum yönetimini hallettikten sonra sıra AWS S3 entegrasyonuna geldi. S3'ü uygulamamda kullanmam için ilk önce bir kova oluşturmam gerekti. Kovalar, S3'te depolanan veriler için konteynerlerdir. Oluşturduğum kovaya veri doldurmam için kovaya ait bilgileri kaydettim. AWS SDK'sını kullanarak S3'ü uygulamaya entegre ettim.

Cognito ve S3 hizmetlerini birlikte kullanarak uygulamaya kullanıcı bazlı dosya yükleme ve kullanıcının dosyalarını listeleme gibi özellikler kattım.

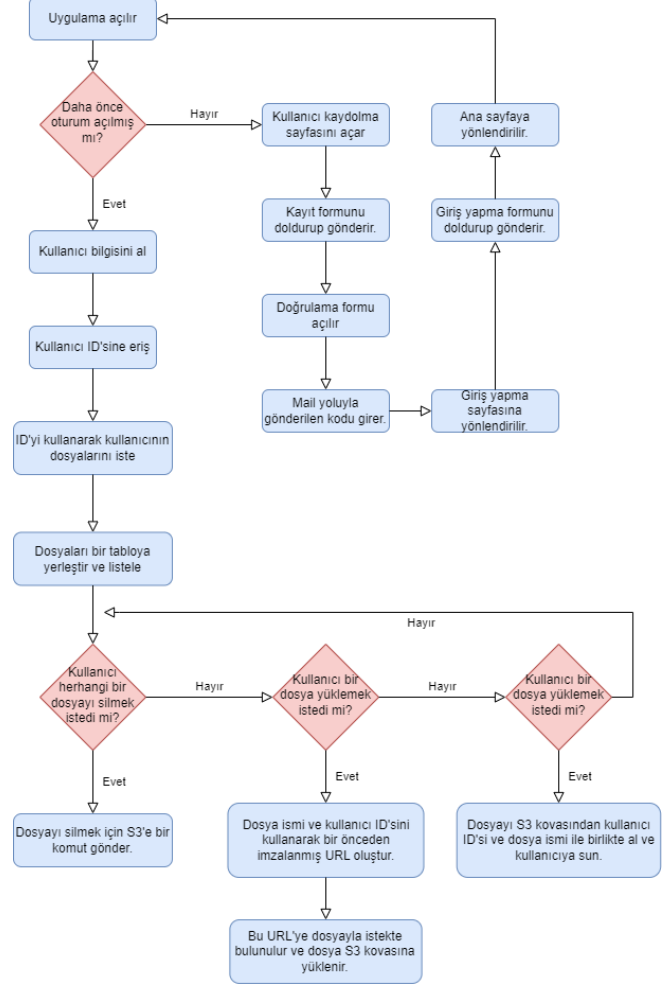
## V. MİMARİ

Kullanıcı ana sayfayı açtığında daha önce oturum açmış mı diye kontrol edilir. Eğer oturum açmamışsa kullanıcının uygulamayı kullanmaya başlaması için kaydolup giriş yapması gerekmektedir. Eğer daha önce oturum açmışsa kullanıcının ID bilgisi alınır. Bu ID bilgisi, kullanıcının depoladığı kendi dosyalarına erişmesi için gereklidir. Bu ID ile birlikte sayfa açılışında S3 istemcisine listeleme komutu verilir. Bu komut sonucunda kullanıcıya ait ne kadar dosya varsa geri döndürülür. Dosyalar ana sayfada bulunan tabloya aktarılır. Kullanıcı tabloda dosya isimlerini ne zaman yüklendiğini ve dosya boyutunu görebilir.

Kullanıcı eğer yeni bir dosya yüklemek isterse bunu kolayca tablonun sağ üstünde bulunan butona basıp açılan diyalogdan formu doldurarak yapabilir. Form doldurulup gönderilince kullanıcının dosyasını güvenli bir şekilde gönderebilmek adına önceden imzalanmış bir URL oluşturulur ve bu URL'ye dosyamız gönderilir. Bu işlemin sonucunda yüklenmek istenen dosya kovaya eklenir.

Kullanıcı eğer bir dosyayı silmek isterse tablodan dosyanın bulunduğu satırı bulup satırın sonundaki silme butonuna basabilir. Bastığında, S3 istemcisine dosya ismiyle bir silme komutu gider ve dosya kovadan kaldırılır.

Kullanıcı eğer tabloda bulunan dosyayı indirmek isterse bu çok basittir, sadece dosya ismine tıklaması yeterlidir.



Şekil 3. Uygulama mimarisinin kısaca özeti.

## KAYNAKLAR

- [1] G. Boss, P. Malladi, D. Quan, L. Legregni, and H. Hall, 'Cloud computing', IBM white paper, vol. 321, pp. 224–231, 2007.
- [2] N. Jain and S. Choudhary, "Overview of virtualization in cloud computing," 2016 Symposium on Colossal Data Analysis and Networking (CDAN), Indore, India, 2016, pp. 1-4, doi: 10.1109/CDAN.2016.7570950.
- [3] Manageengine.com, 2023. <https://www.manageengine.com/network-monitoring/images/bare-metal-hypervisor.jpg> (accessed Dec. 14, 2023).
- [4] Devopsage.com, 2023. <https://www.devopsage.com/wp-content/uploads/2020/01/Type-2-Hypervisor.png> (accessed Dec. 14, 2023).