



**SAKARYA**  
ÜNİVERSİTESİ

**BULUT BİLİŞİM 2.Ödev**

**Danışman : Öğr.Gör. UĞUR ÖZBEK**

**Ahmed Elrimi – B211200552**

## Statik Portföy Web Sitesinin AWS S3 ve CloudFront Üzerinden Yayınlanması

### 1) Proje Özeti

Bu proje kapsamında, kişisel bir portföy (portfolio) web sitesi statik içerik olarak hazırlanmış ve AWS bulut altyapısı üzerinde yayınlanmıştır. Uygulama, HTML + CSS + JavaScript teknolojileriyle geliştirilmiş olup, sunucu taraflı işlem içermeyen statik bir web sitesidir. Yayınlama sürecinde önce Amazon S3 Static Website Hosting kullanılarak site erişilebilir hâle getirilmiş, ardından Amazon CloudFront ile CDN entegrasyonu yapılarak HTTPS desteği sağlanmış ve performans iyileştirilmiştir.

### 2) Projenin Amacı ve Hedefleri

Ödevin amacı; bulut bilişim kavramlarını uygulamalı olarak öğrenmek, bir bulut sağlayıcı platformu üzerinde uygulama dağıtım sürecini deneyimlemek ve bunu raporlayabilmektir. Bu proje özelinde hedefler:

- Statik bir web uygulamasını AWS üzerinde yayınlamak,
- Amazon S3 üzerinde bucket oluşturup yapılandırmak,
- Dosyaları bulut ortamına taşıyarak web sitesi erişimini sağlamak,
- Erişim izinlerini doğru şekilde tanımlamak (public access & bucket policy),
- CloudFront ile HTTPS desteği ve performans optimizasyonu sağlamak,
- Tüm süreci adım adım belgeleyerek raporlamak.

### 3) Uygulama Seçimi (Uygulama Tanımı)

Seçilen uygulama: **Kişisel Portföy Web Sitesi (Static Website)**

Bu uygulama; kişisel tanıtım, beceriler, projeler ve iletişim bilgilerini içeren statik bir web sitesidir.

Uygulama seçiminin başlıca nedenleri:

- Bulut ortamında yayınlama için uygun ve basit bir yapı sunması,
- Sunucu yönetimi gerektirmemesi,
- AWS S3 Static Website Hosting senaryosu ile doğrudan uyumlu olması,
- Dağıtım, erişim kontrolü ve CDN gibi temel bulut kavramlarının pratik olarak gösterilebilmesi.

#### 4) Kullanılan Teknolojiler

##### 4.1 Uygulama tarafı:

- HTML (sayfa yapısı)
- CSS (tasarım ve stil)
- JavaScript (etkileşim ve dinamik davranışlar – client-side)

##### 4.2 Bulut platformu ve servisler:

- AWS (Amazon Web Services)
- Amazon S3 (Static Website Hosting için depolama ve barındırma)
- Amazon CloudFront (CDN + HTTPS + global dağıtım)
- Bucket Policy (s3:GetObject izinleri)
- Public Access ayarları (web erişimi için)

#### 5) Bulut Platformu Seçimi (Neden AWS?)

Bu projede bulut sağlayıcı olarak **AWS** seçilmiştir. Seçim gerekçeleri:

- Yaygın kullanım ve zengin servis ekosistemi,
- **S3 Static Website Hosting** desteği ile statik sitelerin kolay yayınlanabilmesi,
- **CloudFront** ile CDN ve HTTPS desteğinin kolay entegre edilebilmesi,
- Ücretsiz/uygun maliyetli kullanım seçenekleri ve öğrenci projeleri için yeterli kaynaklar sunması.

#### 6) Uygulamanın Bulut Platformuna Taşınması (Adım Adım Dağıtım Süreci)

Aşağıda uygulamanın AWS üzerinde yayınlanması adım adım açıklanmıştır. Her adım ilgili ekran görüntüsü ile belgelenmiştir.

##### 6.1) Amazon S3 Bucket Oluşturma

İlk olarak Amazon S3 üzerinde web sitesi dosyalarını barındırmak için yeni bir bucket oluşturulmuştur.

##### Amazon S3 Üzerinde Yeni Bir Bucket Oluşturulması

Bu adımda AWS Management Console üzerinden S3 servisine girilerek bucket oluşturma ekranı açılmış, bölge (Region) olarak Europe (Stockholm – eu-north-1) seçilmiş ve bucket adı küresel isimlendirme

kurallarına uygun şekilde belirlenmiştir. ACLs disabled (recommended) seçilerek nesne sahipliği ve erişim kontrolü politika tabanlı şekilde yönetilmiştir.

**Create bucket** [Info](#)

Buckets are containers for data stored in S3.

**General configuration**

**AWS Region**  
Europe (Stockholm) eu-north-1

**Bucket type** [Info](#)

☒ **General purpose**  
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

☐ **Directory**  
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

**Bucket name** [Info](#)  
mywebsite-profile  
Bucket names must be 3 to 63 characters and unique within the global namespace. Bucket names must also begin and end with a letter or number. Valid characters are a-z, 0-9, periods (.), and hyphens (-). [Learn more](#)

**Copy settings from existing bucket - optional**  
Only the bucket settings in the following configuration are copied.  
[Choose bucket](#)  
Format: s3://bucket/prefix

**Object Ownership** [Info](#)  
Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

**Object Ownership**

☒ **ACLs disabled (recommended)**  
All objects in this bucket are owned by this account. Access to this bucket and its objects is specified using only policies.

☐ **ACLs enabled**  
Objects in this bucket can be owned by other AWS accounts. Access to this bucket and its objects can be specified using ACLs.

**Object Ownership**  
Bucket owner enforced

## 6.2) Public Access Ayarlarının Yapılandırılması

Statik web sitesinin internet üzerinden erişilebilir olması için bucket'ın public erişim ayarları kontrol edilmiştir.

### Amazon S3 Bucket için Public Access (Genel Erişim) Ayarlarının Yapılandırılması

Bu aşamada “Block Public Access” ayarları incelenmiş ve statik web barındırma senaryosu için gerekli olabilecek genel erişim durumu değerlendirilmiştir. AWS uyarıları dikkate alınarak, web sitesinin yayınlanabilmesi için erişim engelleyebilecek seçenekler yönetilmiştir.

**Block Public Access settings for this bucket**

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to this bucket and its objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to this bucket or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

☒ **Block all public access**  
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

☐ **Block public access to buckets and objects granted through new access control lists (ACLs)**  
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

☐ **Block public access to buckets and objects granted through any access control lists (ACLs)**  
S3 will ignore all ACLs that grant public access to buckets and objects.

☐ **Block public access to buckets and objects granted through new public bucket or access point policies**  
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

☐ **Block public and cross-account access to buckets and objects through any public bucket or access point policies**  
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

**Turning off block all public access might result in this bucket and the objects within becoming public**  
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

☒ I acknowledge that the current settings might result in this bucket and the objects within becoming public.

**Bucket Versioning**  
Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

**Bucket Versioning**

☒ **Disable**

☐ **Enable**

### 6.3) Versioning, Tags ve Default Encryption Ayarları

Bucket'in opsiyonel ayarları yapılandırılmıştır.

#### Amazon S3 Bucket için Sürümleme (Versioning), Etiketleme ve Şifreleme Ayarlarının Yapılandırılması

Bu adımda Bucket Versioning devre dışı bırakılmıştır (statik site için zorunlu değildir). Tags kısmı küçük ölçekli proje kapsamında kullanılmamıştır. Default Encryption bölümünde SSE-S3 seçilerek bucket'a yüklenecek nesnelerin sunucu tarafında otomatik şifrlenmesi sağlanmıştır.

The screenshot shows the 'Create bucket' page in the AWS Management Console. The page is divided into three main sections for configuration:

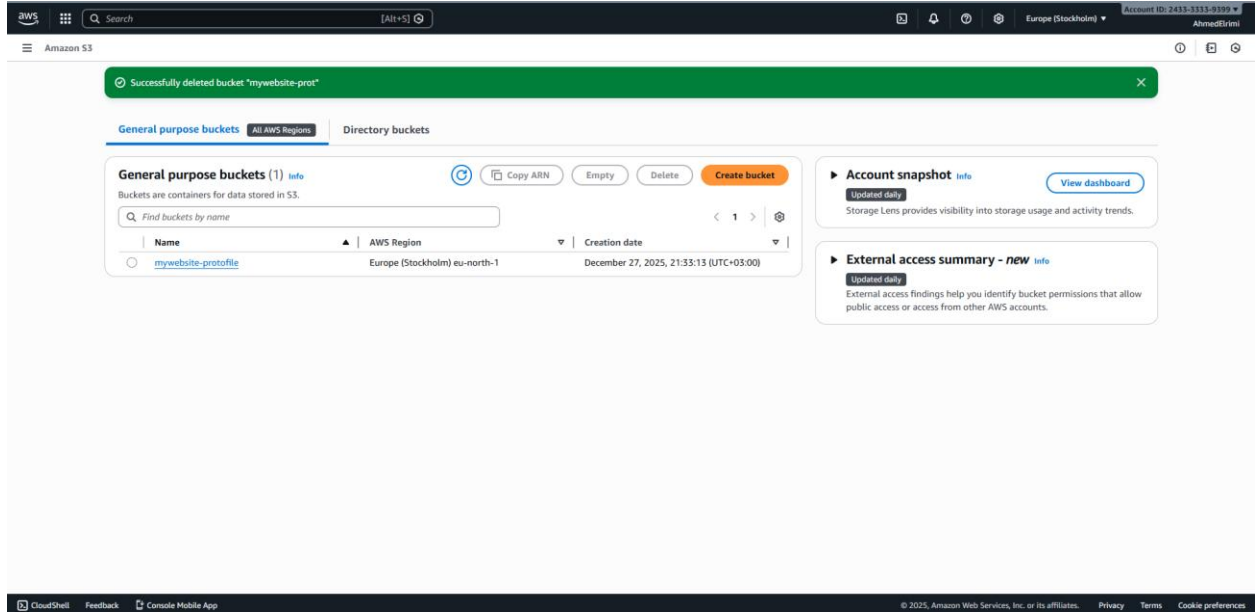
- Bucket Versioning:** A section titled 'Bucket Versioning' with a description: 'Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)'. Below this, there are two radio buttons: 'Disable' (selected) and 'Enable'.
- Tags - optional:** A section titled 'Tags - optional' with a description: 'You can use bucket tags to analyze, manage and specify permissions for a bucket. [Learn more](#)'. Below this, there is a blue box with a warning icon and text: 'You can use s3:ListTagsForResource, s3:TagResource, and s3:UntagResource APIs to manage tags on S3 general purpose buckets for access control in addition to cost allocation and resource organization. To ensure a seamless transition, please provide permissions to s3:ListTagsForResource, s3:TagResource, and s3:UntagResource actions. [Learn more](#)'. Below this, it says 'No tags associated with this bucket.' and there is a button 'Add new tag'. Below the button, it says 'You can add up to 50 tags.'
- Default encryption:** A section titled 'Default encryption' with a description: 'Server-side encryption is automatically applied to new objects stored in this bucket.' Below this, there is a section 'Encryption type' with three radio buttons: 'Server-side encryption with Amazon S3 managed keys (SSE-S3)' (selected), 'Server-side encryption with AWS Key Management Service keys (SSE-KMS)', and 'Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)'. Below this, there is a section 'Bucket Key' with a description: 'Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. [Learn more](#)'. Below this, there are two radio buttons: 'Disable' and 'Enable' (selected).

### 6.4) Bucket'in Başarıyla Oluşturulduğunun Doğrulanması

Bucket oluşturma işlemi tamamlanmış ve konsolda listelendiği doğrulanmıştır.

#### Amazon S3 Bucket'in Başarıyla Oluşturulması

Bu ekranda bucket'in ilgili bölgede aktif olarak oluşturulduğu ve dosya yüklemeye hazır olduğu görülmektedir. Bu aşama, dağıtım ortamının hazırlandığını göstermektedir.

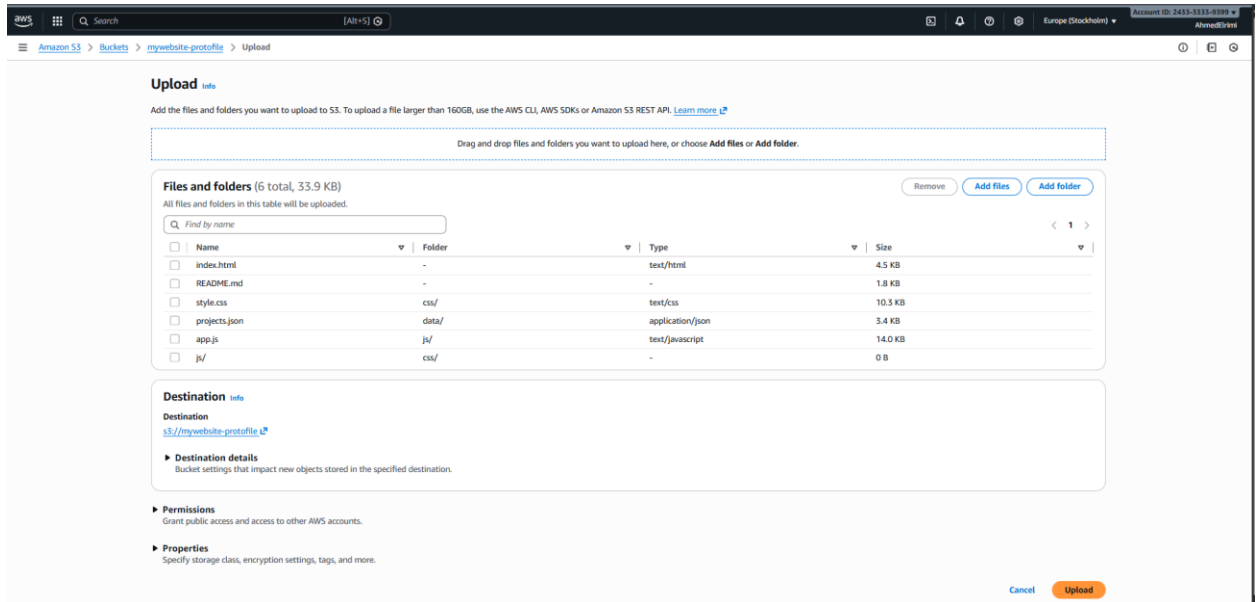


## 6.5) Web Sitesi Dosyalarının S3'e Yüklmesi

Portföy sitesine ait dosyalar bucket içine yüklenmiştir.

### Statik Web Sitesi Dosyalarının Amazon S3 Bucket'a Yüklmesi

Bu adımda Upload ekranı üzerinden index.html, CSS ve JavaScript dosyaları seçilerek bucket içine aktarım başlatılmıştır. Böylece uygulama bulut ortamına taşınmıştır.



## 6.6) Yükleme İşleminin Başarıyla Tamamlanması

Yükleme işleminin hatasız tamamlandığı doğrulanmıştır.

### Web Sitesi Dosyalarının Amazon S3 Bucket’a Başarıyla Yükleneceği

Bu ekranda tüm dosyaların “Succeeded” durumu ile yüklendiği ve “Failed” dosya bulunmadığı görülmektedir. Bu durum, uygulama içeriklerinin eksiksiz şekilde buluta taşındığını doğrular.

The screenshot displays the AWS S3 console interface. At the top, a green banner indicates "Upload succeeded" with a message: "For more information, see the Files and folders table." Below this, the "Upload: status" section shows a summary of the upload process. The destination is "s3://mywebsite-prototype". The summary table shows that 6 files (33.9 KB) were successfully uploaded (100.00%), and 0 files (0 B) failed (0%).

| Summary                  |                            |
|--------------------------|----------------------------|
| Destination              | Success                    |
| s3://mywebsite-prototype | 6 files, 33.9 KB (100.00%) |

Below the summary, the "Files and folders" section is active, showing a table of the uploaded files and folders. The table has columns for Name, Folder, Type, Size, Status, and Error. All files are listed as "Succeeded".

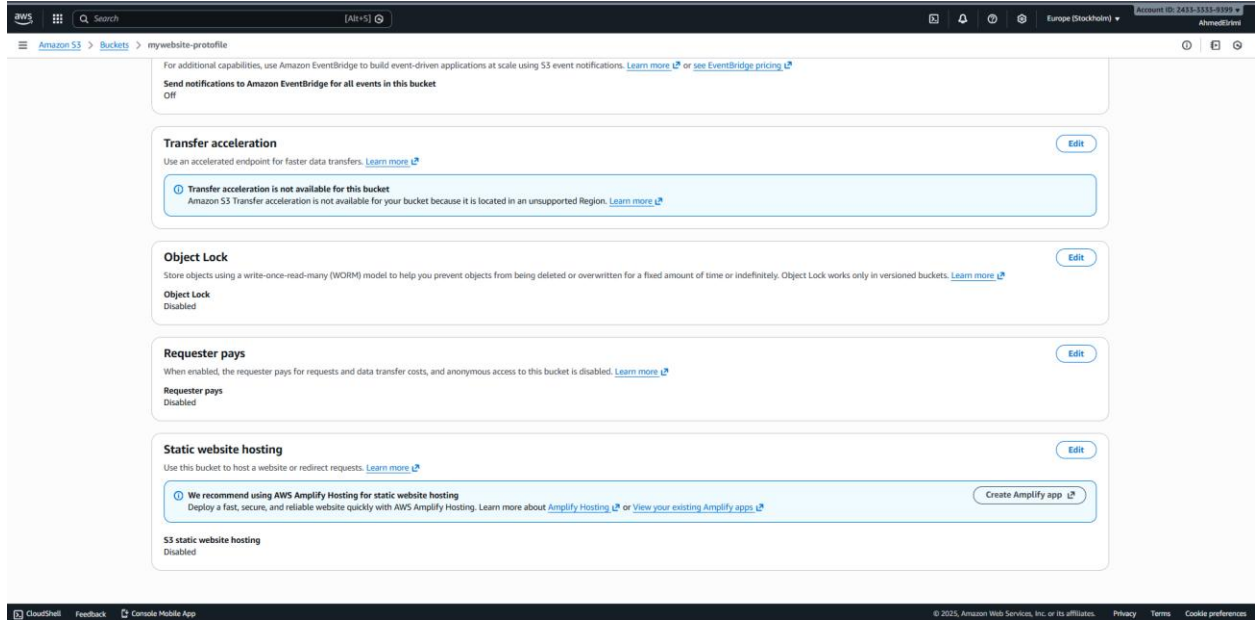
| Name         | Folder | Type             | Size    | Status    | Error |
|--------------|--------|------------------|---------|-----------|-------|
| index.html   | -      | text/html        | 4.5 KB  | Succeeded | -     |
| README.md    | -      | -                | 1.8 KB  | Succeeded | -     |
| style.css    | css/   | text/css         | 10.3 KB | Succeeded | -     |
| project.json | data/  | application/json | 3.4 KB  | Succeeded | -     |
| app.js       | js/    | text/javascript  | 14.0 KB | Succeeded | -     |
| js/          | css/   | -                | 0 B     | Succeeded | -     |

## 6.7) Static Website Hosting’in Etkinleştirilmesi

Bucket’ın web sitesi olarak yayın yapabilmesi için Static Website Hosting ayarı açılmıştır.

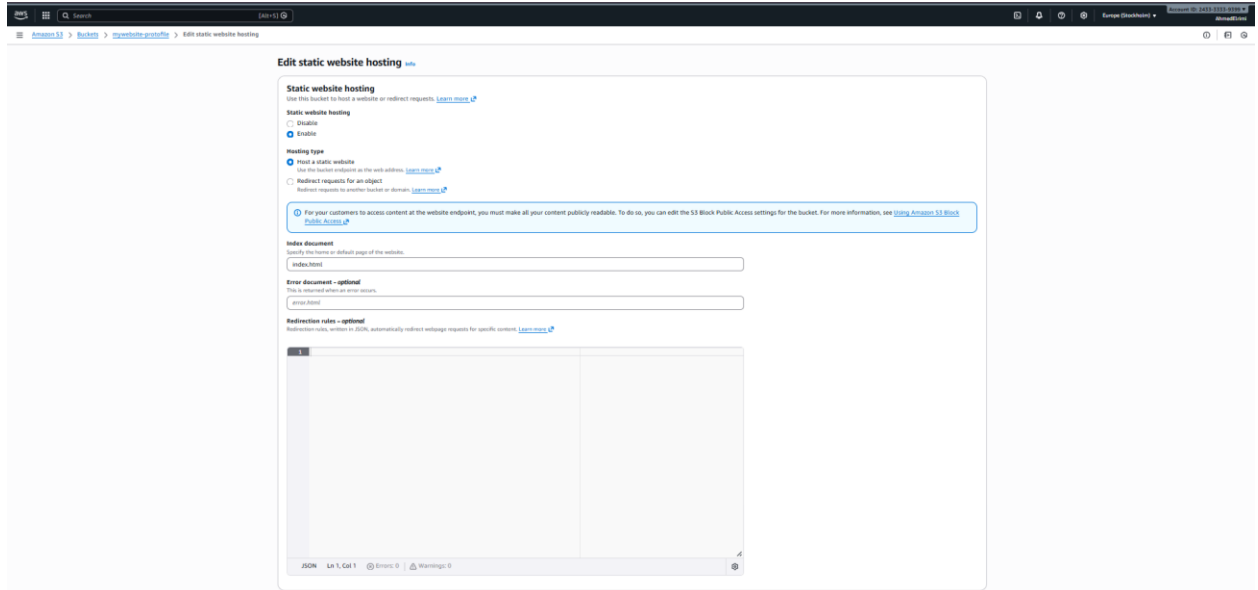
### Amazon S3 Bucket için Static Website Hosting Özelliğinin Yapılandırılması

Properties bölümünde Static Website Hosting kısmına girilerek web barındırma özelliğinin etkinleştirileceği yapılandırma adımı başlatılmıştır.



## Amazon S3 Static Website Hosting Ayarlarının Etkinleştirilmesi

“Host a static website” seçilerek index.html ana sayfa (Index document) olarak tanımlanmıştır. Böylece S3 bucket, statik web sitesi sunabilecek şekilde yapılandırılmıştır.



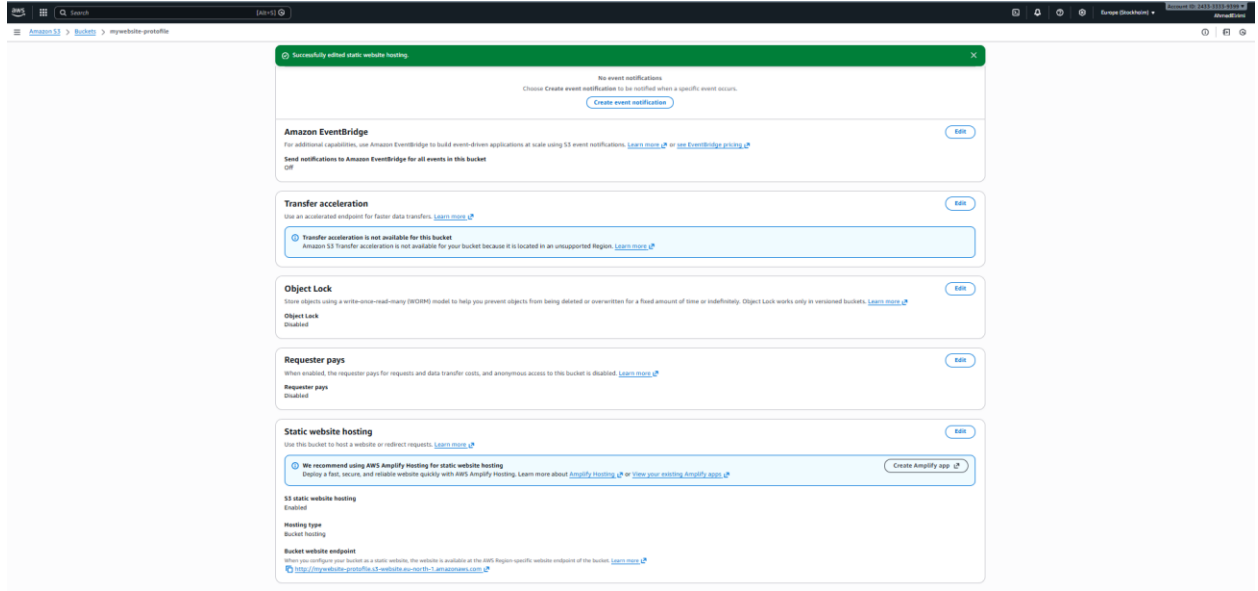


## 6.8) Website Endpoint'in Oluşturulması ve Yayın Doğrulaması

Hosting aktif edildikten sonra AWS tarafından endpoint oluşturulmuştur.

### Amazon S3 Static Website Hosting'in Başarıyla Etkinleştirilmesi ve Website Endpoint'in Oluşturulması .

Bu ekranda S3 website endpoint'in olduğu görülmektedir. Bu endpoint üzerinden siteye erişim sağlanarak yayının başarılı olduğu doğrulanmıştır.

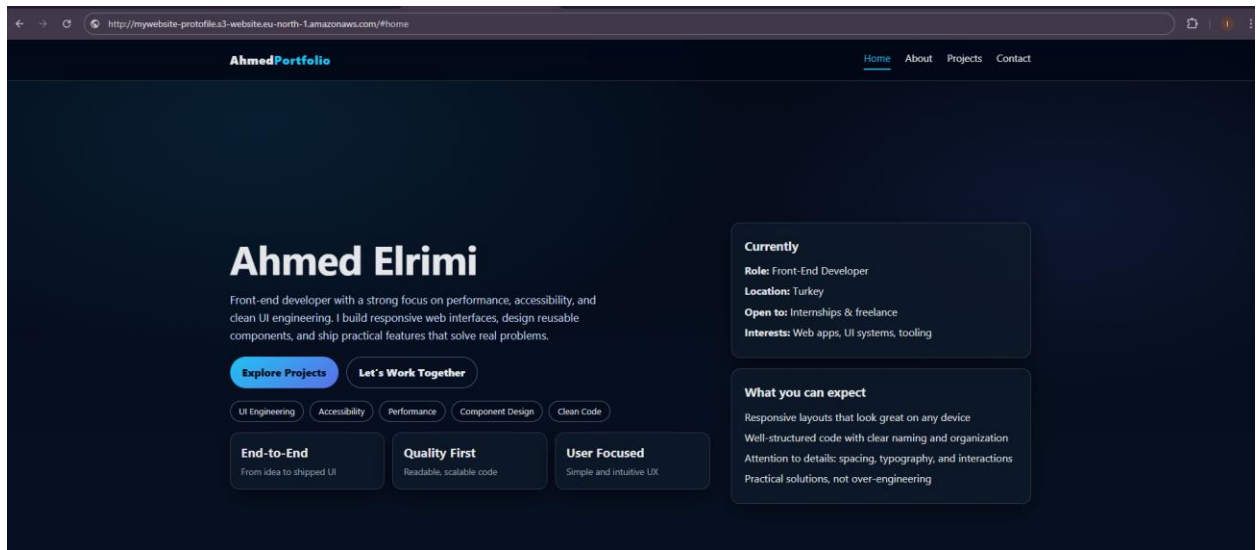
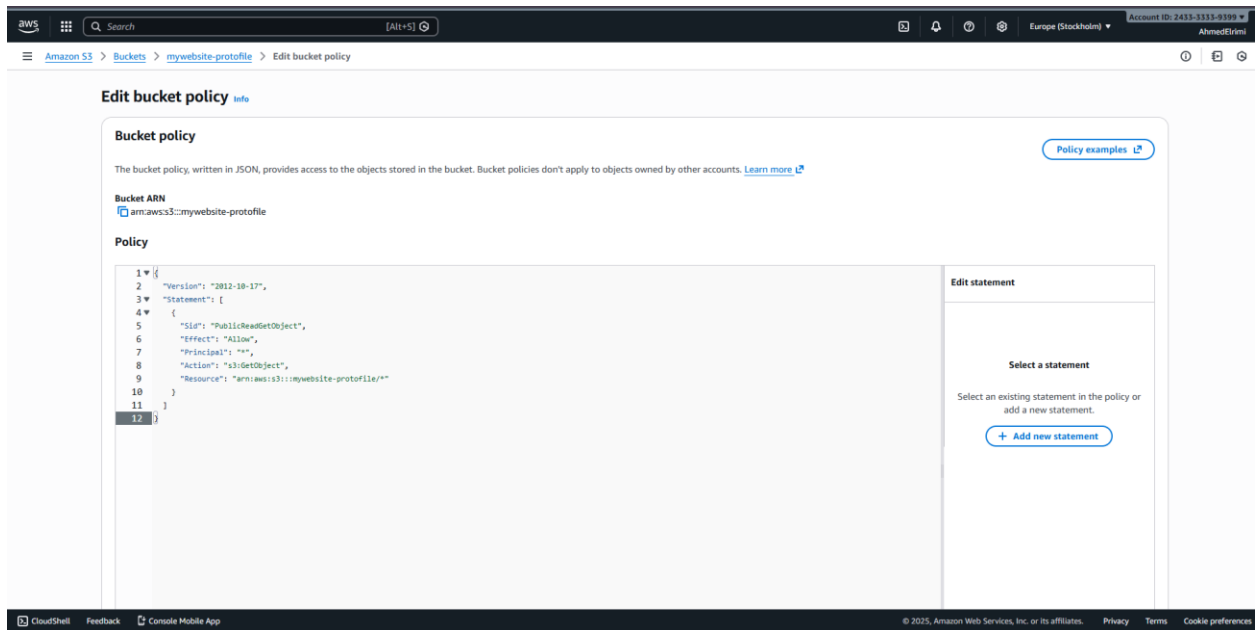


## 6.9) Bucket Policy Tanımlanması (Public Read)

Web tarayıcılarının dosyalara erişebilmesi için bucket policy yazılmıştır.

### Amazon S3 Bucket Policy ile Genel Okuma (Public Read) Erişiminin Tanımlanması

Bu adımda JSON tabanlı bucket policy ile s3:GetObject yetkisi tanımlanarak bucket içindeki nesnelerin okunabilmesi sağlanmıştır. Böylece index.html, CSS ve JS dosyalarına internet üzerinden erişim mümkün hâle gelmiştir.



## 7) Gelişmiş Yapılandırma: CloudFront ile HTTPS ve Performans Artırımı

S3 static website hosting ile yayın sağlandıktan sonra, projeye ek iyileştirme olarak CloudFront kullanılmıştır. CloudFront kullanımı ile:

- HTTPS üzerinden güvenli erişim,
- CDN ile global edge noktalarından daha hızlı teslim,
- Daha iyi performans ve ölçeklenebilirlik hedeflenmiştir.

### 7.1) CloudFront Dağıtım Oluşturma Sürecinin Başlatılması

#### Amazon CloudFront Dağıtımının Oluşturulmasına Başlanması

CloudFront “Create distribution” adımı ile yeni bir dağıtım oluşturma süreci başlatılmıştır. Dağıtım tipi olarak “Single website or app” seçilerek tek bir statik site için uygun yapı tercih edilmiştir.

Step 1: Get started

Step 2: Specify origin

Step 3: Enable security

Step 4: Get TLS certificate

Step 5: Review and create

### Get started

Connect your websites, apps, files, video streams, and other content to CloudFront. We optimize the performance, reliability, and security for your web traffic.

#### Distribution options

**Distribution name**  
Name will be stored as a tag on the resource. You can change the name, or more tags, later.  
portfolio website

**Description - optional**  
It is personal website

**Distribution type**

☒ Single website or app  
Choose if each website or application will have a unique configuration.

☐ Multi-tenant architecture - New  
Choose when you have multiple domains that need to share configurations. This is a common architecture for SaaS providers.

#### Domain

**Route 53 managed domain - optional**  
Enter a domain that's already registered with Route 53 in your AWS account. CloudFront will provision a TLS certificate for you. If you have a domain from a different DNS provider, skip this step and configure your domain later.  
Check domain

Tags - optional

Cancel Next

### 7.2) Origin Olarak S3'ün Seçilmesi

#### Amazon CloudFront Dağıtımını için Origin Olarak Amazon S3'ün Tanımlanması

Bu adımda origin türü Amazon S3 olarak seçilmiş ve içerik kaynağı, daha önce oluşturulan S3 bucket ile ilişkilendirilmiştir. Önerilen origin ve cache ayarları kullanılarak standart ve doğru yapılandırma sağlanmıştır.

Step 1: Get started  
Step 2: Specify origin  
Step 3: Enable security  
Step 4: Review and create

### Specify origin

Your origin is where your content (such as a website or app) lives. CloudFront works with AWS-based origins and origins hosted on other cloud providers.

**Origin type**

☒ Amazon S3  
Deliver static assets like files and images, statically generated websites or single page applications (SPAs).

☐ Elastic Load Balancer  
Deliver applications hosted behind ELB such as dynamic websites, web services, and APIs.

☐ API Gateway  
Deliver API endpoints for REST APIs hosted on API Gateway.

☐ VPC origin  
Deliver applications and content hosted within private VPCs, such as EC2 instances and Application Load Balancers.

☐ Other  
Refer to any AWS or non-AWS origin through its publicly resolvable URL.

**Origin**

**S3 origin**  
Choose an AWS origin, or enter your origin's domain name. [Learn more](#)

mywebsite-profiles.s3.eu-north-1.amazonaws.com [Browse S3](#)

**Origin path - optional**  
The directory path within your origin where your content is stored. [Learn more](#)

/path

**Settings** [info](#)  
CloudFront provides default origin and cache settings based on what origin you selected. [View default settings for S3](#)

**Allow private S3 bucket access to CloudFront** [info](#)  
CloudFront will obtain your S3 bucket policy to allow CloudFront to access your S3 bucket. The policy allows CloudFront to access the bucket only when the request is on behalf of the CloudFront distribution that contains the S3 origin.

☒ Allow private S3 bucket access to CloudFront - Recommended

**Origin settings**  
Origin settings control how CloudFront connects to the specified origin.

☒ Use recommended origin settings ☐ Customize origin settings

**Cache settings**  
Cache settings determine when CloudFront serves cached content and when it fetches new content from the origin.

☒ Use recommended cache settings tailored to serving S3 content ☐ Customize cache settings

### 7.3) Güvenlik (WAF) Ayarları

#### Amazon CloudFront Dağıtımı için Güvenlik (WAF) Ayarlarının Yapılandırılması

Bu aşamada AWS WAF seçeneği değerlendirilmiştir. Proje statik bir portföy sitesi olduğu ve kullanıcı girdisi/dinamik işlem içermediği için WAF etkinleştirilmemiştir. Böylece gereksiz maliyet ve karmaşıklık önlenmiştir.

Step 1: Get started  
Step 2: Specify origin  
Step 3: Enable security  
Step 4: Review and create

### Enable security

**Web Application Firewall (WAF)** [info](#)

☐ Enable security protections  
Keep your application secure from the most common web threats and security vulnerabilities using AWS WAF. Blocked requests are stopped before they reach your web servers.

☒ Do not enable security protections  
Select this option if your application does not need security protections from AWS WAF.

[Cancel](#) [Previous](#) [Next](#)

### 7.4) Yapılandırmaların Gözden Geçirilmesi ve Dağıtımın Oluşturulması

#### Amazon CloudFront Dağıtımı İçin Yapılandırmaların Gözden Geçirilmesi ve Oluşturulması

Dağıtım oluşturulmadan önce origin, cache ve security ayarları kontrol edilmiştir. Ardından “Create distribution” ile dağıtım oluşturma işlemi başlatılmıştır.

Step 1Get startedStep 2Specify originStep 3Enable securityStep 4Review and create

Review and create

General configuration

Distribution nameportfolio websiteDescriptionit is personal websiteBillingPay-as-you-go (\$0/month)

Origin

Because you granted CloudFront access to your origin, CloudFront can write and update S3 bucket policies that restrict access to your S3 origin to CloudFront.

S3 originmywebsite-prototype.s3.eu-north-1.amazonaws.comOrigin path-Grant CloudFront access to originYesEnable Origin ShieldNoConnection attempts3Connection timeout10

Cache settings

CloudFront will apply default cache settings tailored to serving content from a S3 origin. You can customize settings after you create your distribution.

Security

Security protectionsNoneUse monitor modeNoUse existing WAF configurationNo

CancelPreviousCreate distribution

## 7.5) CloudFront Dağıtımının Başarıyla Oluşturulması

### Amazon CloudFront Dağıtımının Başarıyla Oluşturulması ve HTTPS Üzerinden Erişimin Sağlanması

Bu ekranda CloudFront dağıtımının başarıyla oluşturulduğu ve bir distribution domain name üretildiği görülmektedir. Bu alan adı üzerinden siteye **HTTPS** ile erişim sağlanmış; CDN sayesinde içerik teslim performansı iyileştirilmiştir.

Successfully created new distribution.

portfolio website

View metrics

Details

Distribution domain named26ggc6ec04k1a.cloudfront.netBillingPay-as-you-goSwitch to a planARNamawscloudfront:24333339399:distribution/EZDE4XJCLHMOF5Last modifiedDecember 27, 2025 at 7:06:01 PM UTC

GeneralSecurityOriginsBehaviorsError pagesInvalidationsLoggingTags

Settings

Nameportfolio websiteDescriptionit is personal websitePrice classUse all edge locations (best performance)Supported HTTP versionsHTTP/2, HTTP/1.1, HTTP/1.0Alternate domain namesAdd domainStandard loggingOffCookie loggingOffDefault root object-

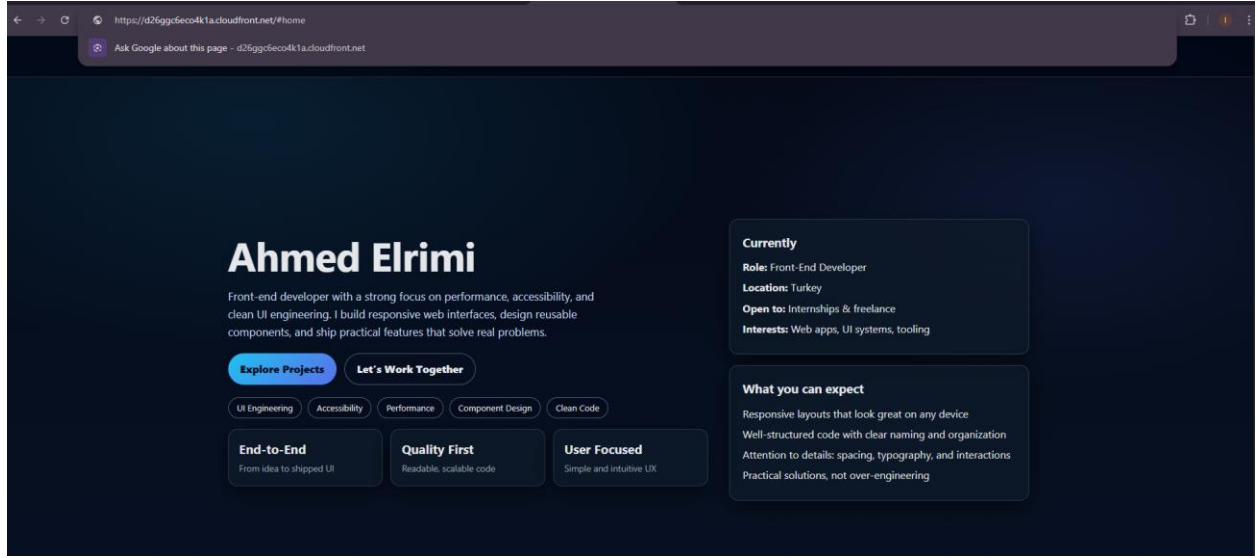
Edit

Continuous deployment info

Create staging distribution

CloudShellFeedbackConsole Mobile App

© 2025, Amazon Web Services, Inc. or its affiliates. PrivacyTermsCookie preferences



## 8) Uygulama Mimari Açıklaması

Bu projenin genel mimarisi aşağıdaki gibidir:

- Kullanıcı (Web Tarayıcısı)
  - CloudFront (CDN + HTTPS)
  - Amazon S3 Bucket (Static Website Dosyaları: HTML/CSS/JS)

S3, içeriğin depolandığı ve kaynak (origin) olarak görev yapan katmandır. CloudFront ise bu içeriği edge lokasyonlar üzerinden kullanıcıya daha hızlı ve güvenli (HTTPS) bir şekilde sunar.

## 9) Karşılaşılan Zorluklar ve Çözümler

Dağıtım sürecinde dikkat edilmesi gereken temel noktalar:

- **Erişim izinleri:** Statik web sitesi için public erişim ve bucket policy doğru tanımlanmalıdır. Aksi halde 403 AccessDenied hataları görülebilir.
- **Static Website Hosting yapılandırması:** Index document olarak doğru dosya (index.html) tanımlanmalıdır.
- **CloudFront geçişi:** CloudFront oluşturulduktan sonra yayının tam aktif olması için dağıtımın “Deployed” durumuna gelmesi beklenmeli ve cache davranışları dikkate alınmalıdır.

## 10) Öğrenilen Dersler

Bu proje ile birlikte aşağıdaki kazanımlar elde edilmiştir:

- AWS S3 üzerinde bucket oluşturma ve temel yapılandırma,
- Statik web sitesi barındırma (Static Website Hosting) mantığı,
- Public access ve bucket policy ile erişim kontrolü,
- CloudFront CDN kavramı ve HTTPS ile güvenli yayın,
- Bulut servislerinin performans ve güvenlik açısından nasıl birlikte kullanılabileceği.

## 12) Olası İyileştirmeler

Proje gelecekte aşağıdaki şekillerde geliştirilebilir:

- Özel alan adı (custom domain) bağlama (Route 53 veya dış DNS ile),
- CloudFront üzerinde daha gelişmiş cache politikaları,
- Logging (S3 access logs / CloudFront logs) etkinleştirme,
- CI/CD ile otomatik deploy (GitHub Actions ile S3 sync),
- Güvenlik için WAF kuralları (ihtiyaç halinde).