KLAVYE HIZ TESTİ UYGULAMASI VE AWS SUNUCU ENTEGRASYONU

*Ad Soyad:Furkan Esad Uzun*  
*Öğrenci No: 211307059*  
*Okul:Kocaeli Üniversitesi*Kocaeli, Türkiye

Proje Drive Linki: <https://drive.google.com/file/d/1F-RMqSsga62o9aXvCgWTsT2rNqLebdTF/view?usp=sharing>

Bu rapor, HTML, CSS ve JavaScript kullanarak geliştirilen bir klavye hız testi uygulamasının, Amazon Web Services (AWS) üzerindeki bir Windows sunucusuna nasıl entegre edildiğini açıklamaktadır. Uygulama, kullanıcıların klavye hızlarını test etmelerine olanak tanıyan interaktif bir arayüze sahiptir. Uygulama, belirli bir metin paragrafını kullanıcılara gösterir ve kullanıcıların bu metni hızlı ve doğru bir şekilde yazmalarını ölçer. Ayrıca, kullanıcıların performansını değerlendiren istatistikleri sunar. Uygulama, kullanıcıların sonuçlarını kaydederek, performanslarını izlemelerine ve iyileştirmelerine olanak tanır.

Anahtar kelimeler - HTML, CSS, JavaScript, Klavye Hız Testi, AWS, Windows Sunucu

# giriş

Teknolojinin sürekli evrim geçirdiği günümüzde, klavye hızının bireylerin bilgisayar kullanımı üzerindeki etkisi giderek artmaktadır. Hızlı ve doğru yazma becerisi, birçok profesyonel alanda kritik bir avantaj sağlamaktadır, özellikle de bilgi işlem, yazılım geliştirme ve diğer teknoloji odaklı sektörlerde. Bu bağlamda, klavye hızını değerlendirmek ve iyileştirmek isteyen bireyler için interaktif bir klavye hız testi uygulamasının geliştirilmesi önemli bir adımdır.

Bu rapor, kullanıcıların klavye hızlarını etkili bir şekilde ölçmelerine ve geliştirmelerine olanak tanıyan bir web uygulamasının detaylı bir açıklamasını sunmaktadır. Uygulama, HTML, CSS ve JavaScript teknolojilerini kullanarak geliştirilmiş olup, kullanıcıların hız testi sonuçlarını takip etmelerini sağlamak amacıyla Amazon Web Services (AWS) üzerindeki bir Windows sunucusuna entegre edilmiştir. Bu entegrasyon, uygulamanın dünya çapında geniş bir kullanıcı kitlesine kesintisiz ve güvenilir bir deneyim sunmasına olanak tanımaktadır. Bu rapor, uygulamanın teknik altyapısını, AWS entegrasyonunu ve kullanıcı deneyimini detaylandıracak, ayrıca uygulamanın getirdiği avantajları ele alacaktır.

# Teknik Altyapı ve Kod Mimarisi

Uygulama, HTML, CSS ve JavaScript dillerini kullanarak geliştirilmiştir. Temelde, kullanıcıya bir metin paragrafı gösterilir ve bu metni hızlı ve doğru bir şekilde yazmasını ölçmek üzere tasarlanmıştır. HTML, sayfanın temel yapısını belirler ve kullanıcı arayüzünü düzenler. CSS, arayüzü stilize eder ve kullanıcıya hoş bir görsel deneyim sunar. JavaScript ise, uygulamanın dinamik ve etkileşimli özelliklerini yönetir. Kullanıcı metni girdikçe, JavaScript kodları kullanıcının girişini anlık olarak değerlendirir ve geri bildirimde bulunur.

AWS sunucu entegrasyonu, uygulamanın ölçeklenebilir ve güvenilir bir şekilde çalışmasını sağlamak amacıyla gerçekleştirilmiştir. Uygulama, AWS üzerindeki bir Windows sunucusu üzerinde barındırılmaktadır. Sunucu, uygulamanın depolama ihtiyaçlarını karşılamak ve yüksek performans sağlamak üzere özelleştirilmiştir. Uygulama, kullanıcıların test sonuçlarını güvenli bir şekilde depolayabilmek ve kullanıcılara kişisel bir deneyim sunmak adına AWS üzerindeki depolama servislerini etkin bir şekilde kullanır.

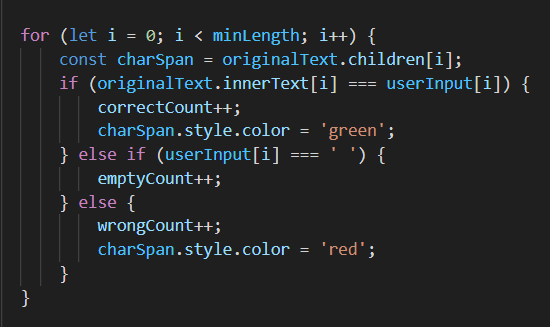
# Kod açıklamaları

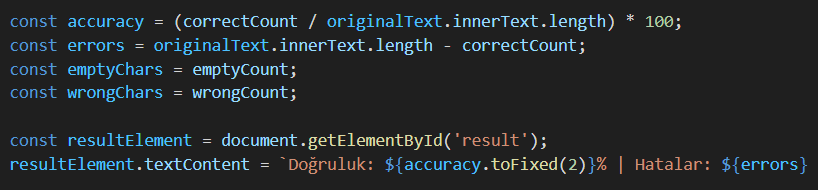
Uygulamanın HTML yapısı, sayfanın temel görsel bileşenlerini oluşturur. Ana başlık, metin gösterimi, metin giriş alanı ve buton gibi elemanlar, kullanıcının etkileşime geçtiği arayüzü tanımlar. Özellikle metin paragrafının her bir karakteri, <span> elementleri içinde yer alır. Bu, JavaScript kodları tarafından kullanıcının metin girişi ile eşleşen karakterleri renklendirebilmesi için bir zemin oluşturur. Bu yapı, uygulamanın gerçek zamanlı olarak kullanıcı girişiyle etkileşimde bulunmasını sağlar.

CSS stillemesi, sayfanın genel görünümünü belirler. Arka plan resmi, sayfanın atmosferini oluştururken, metin girişi alanının ve butonun stili, kullanıcı etkileşimini anlamlandırmak ve hoş bir görsel deneyim sunmak için özelleştirilmiştir. Renk seçimleri ve gölge efektleri, kullanıcının uygulama üzerindeki etkileşimini vurgulamak ve genel estetik dengeyi korumak için dikkatlice seçilmiştir.

Hata sayısı (wrongCount) ve boş karakter sayısı (emptyCount) her karakter karşılaştırıldığında güncellenir.

Karakterler doğru ise, yeşil renkte gösterilir; boş ise, sayıları artırılır; yanlış ise, kırmızı renkte gösterilir.



 ‘accuracy’ değişkeni, doğru karakterlerin toplam karakter sayısına oranını temsil eder. Sonuçlar, resultElement üzerinde kullanıcıya gösterilir.

 saveResults() fonksiyonu, kullanıcıdan alınan bilgileri bir metin dosyasına formatlı bir şekilde ekler.

formatTime() fonksiyonu, verilen saniye cinsinden süreyi dakika ve saniye cinsine çevirir.

generateRandomParagraph() fonksiyonu, belirli bir kelime havuzundan istenilen sayıda kelime içeren rastgele bir paragraf oluşturur.



AWS sunucu entegrasyonu, uygulamanın güçlü bir altyapıya sahip olmasını sağlar. AWS üzerindeki bir Windows sunucusu, uygulamanın dünya çapındaki kullanıcılara hızlı ve güvenilir bir hizmet sunabilmesini sağlar. Sunucu, uygulamanın depolama ve performans gereksinimlerini karşılamak üzere özelleştirilmiştir. AWS'nin depolama hizmetleri, kullanıcıların performans sonuçlarını güvenli bir şekilde depolamak ve erişmek için kullanılır.

# karşılaştığım zorluklar

Birincil zorluk, AWS sunucu entegrasyonunda yaşandı. AWS'nin karmaşık yapıları ve özelleştirilebilir hizmetleri, başlangıçta uygulamanın bu platforma sorunsuz bir şekilde entegre edilmesini engelledi. Bu noktada, AWS belgeleri ve topluluk forumları üzerinden kapsamlı bir araştırma ve deneme-yanılma süreci gerçekleştirildi. Entegrasyon sürecinde yaşanan bu zorluklar, sonunda uygulamanın AWS üzerinde güçlü bir şekilde çalışmasını sağlayan özelleştirilmiş bir çözümle aşıldı.

##### References

Bana yardımcı olan videolar:

* <https://www.youtube.com/watch?v=voa0btsVSfk&pp=ygUMYXdzIHdlYiBhcHAg>
* <https://www.youtube.com/watch?v=7m_q1ldzw0U&pp=ygUMYXdzIHdlYiBhcHAg>
* <https://www.youtube.com/watch?v=sCQwEVhCvTg&pp=ygUMYXdzIHdlYiBhcHAg>