

# T.C. FATİH SULTAN MEHMET VAKIF ÜNİVERSİTESİ

MÜHENDİSLİK FAKÜLTESİ

Merve TANRIKULU

1421221024

1-)Proje Adı

Hangman

## 2-)Proje Konusu

**Harry Potter temalı bir adam asmaca oyunu.**

## 3-)Proje’de Yapılanlar

**Proje CSS,JS ve HTML sayfası olmak üzere 3 sayfadan oluşmaktadır.**

**CSS sayfasında ekran boyutlarının responsive olabilmesi için boyutlar yüzdelerle ifade edilmiştir. Bu sayfada animasyon, asılan adam, kullanılabilecek harfler, ayrıca klavyeden girilen harfleri de görüntüleyen bir kısım oluşturulmuş ve bunların düzenlenmeleri yapılmıştır.**

**HTML sayfasında animasyonda kullandığım resimler ve yazılar yer almaktadır. Proje de kullanmış olduğum isteğe bağlı açıp kapanabilen müzikte bu sayfada mevcuttur. Sayfadaki butonlar ve adam asmacanın resmide yine bu sayfada bulunmaktadır.**

JS sayfasında kelimeleri dosyadan çekebilmek için readData() fonksiyonu bulunmaktadır. Oyunda kullanılan fonksiyonlar Hangman adlı bir nesne içinde yazılmıştır. İçinde öncelikle sayfa açıldığında gelmesi gereken fonksiyonlar initilaize() ‘ın içine yazılmıştır ve randomDisplayphrase(), showDisplayPhrase(), showAvailableLetters() fonksiyonlarını çağırır. Ayrıca bu fonksiyonda sayfa yüklendiğinde kelimelerin de diziye atılabilmesi için bir kopyalama işlemi bulunmaktadır. randomDisplayphrase() fonksiyonu Math.random() sınıfını kullanarak kelimeleri diziden okur ve bir rastgele sayı üretir. showDisplayPhrase() fonksiyonu öncelikle üretilen kelimenin boyutu kadar dönerek bir div oluşturur daha sonra bir class ismi verir ve üretin kelimedeki harf sayısı kadar çizgi üretir. Boşluk içinde bir kontrolü vardır ve karakter olarak sayar.

showAvailableLetters() fonksiyonuda bu mantıkta çalışmaktır farklı olarak harfe tıklandığı zaman kelimenin içinde var olup olmadığının kontrolünü yapabilmek için başka bir fonksiyona gönderilir. Kelime değiştiğinde oluşan divlerin silinmesi içinde removeDivs() fonskiyonu bulunmaktadır. Oluşan classlara göre silme işlemi yaptığından class ismi içeren bir parametre alır. letterMatchCheck() fonksiyonu kullanıcı bir harf girmişse kelimenin içinde dolaşmak için bir döngüye girer ve karşılaştırma yapar. Eğer bir uyuşma olmazsa adama bir parça ekler.

phraseMatchCheck() fonskiyonu eğer tahmin edilen bir kelime ise o kelimenin doğru olup olmadığına bakar. Bu iki işlemde yazılan kelime küçük büyük harf problemi yaşanmaması için toUpperCase() fonskiyonu ile kontrol edilmiştir. updateDisplayPhrase() ise letterMatchCheck(), phraseMatchCheck(), removeDivs(), showDisplayPhrase() fonksiyonlarını çağırır çünkü her eklenen harf için farklı bir görüntü oluşmaktadır. showPhrase() fonksiyonu tahmin edilmesi gereken kelimeyi gösterir. animateLetters() fonksiyonu kelime bilindiği ya da bilinemediği takdirde kelimenin ne olduğunu gösterirken bunu animasyon haline çevirmektedir. addGamePiece() fonksiyonu switch case yapısında çalışır bir yanlış kelime sayacı bulunmaktadır sayaç arttıkça case lerin numaralarıda artar ve her seferinde adama bir parça eklenir. resetGame() oyunun sıfırlandığı bir fonksiyondur. onEnter() metodu klavyeden hangi kelimelere basıldığını bulmak için bir keyboard event kullanıldı (e.which || e.keyCode) enter tuşuna basıldığıda (e.which == 13) yine aynı şekilde bir keyboard event ile bulundu ve daha sonra onSubmit() fonksiyonuna gönderildi. onSubmit() fonksyionu hem sayfa üzerinden tıklamanın hem de klavyeden girilen fonksiyonun kontrolünü yaptığı için inputORclickedLetter parametresini aldı. Bu fonksiyon eğer gelen harf veya kelime klavyedense gelen inputu yani yazığımız kelime ya da harfi önce büyük harfe çeviriyor böylece küçük büyük harf sorunu yaşanmasın ve temizliyor diğer kelime için önceki harf veya kelime orda kalmasın. PhrasesClue() fonskiyonu ise kullanıcıya yardımcı olabilmek için kelimeyle alakalı ipuçları çıkarmakta bu fonksiyonda her kelime için ayrı bir animasyon bulunmaktadır.