

**ANKARA ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**



BLM 4531 PROJE RAPORU

BeeFocus – YKS Çalışma Takip Sistemi

İbrahim Çelik

22290065

Enver Bağcı

15.01.2026

Github: <https://github.com/clkibrahim/BeeFocus>

Proje Videosu:

<https://drive.google.com/file/d/10qAwRobKwJlrgELTKM8G5y4CXc6820kJ/view?usp=sharing>

ÖZET

Bu çalışmada, kullanıcıların odaklanarak geçirdikleri çalışma sürelerini kayıt altına almasını ve bu verileri raporlayabilmesini sağlayan BeeFocus uygulaması geliştirilmiştir. Uygulama, çalışma seanslarının başlatılması ve sonlandırılması ile seanslara ait başlangıç/bittiş zamanı, süre ve kategori bilgilerini saklayacak şekilde tasarlanmıştır. Toplanan veriler üzerinden günlük ve haftalık özetler, günlük dağılım grafikleri ve kategori bazlı çalışma istatistikleri üretilerek kullanıcıya sunulmaktadır.

Proje kapsamında, mobil uygulama ve backend API katmanı ile birlikte, hocanın istediği şekilde backend'e bağlı çalışan bir web arayüzü oluşturulmuştur. Web tarafında kullanıcı girişi ve raporlar sayfası yer almaktır; raporlar sayfası backend üzerinden sağlanan verilerle dinamik olarak güncellenmektedir. Sonuç olarak BeeFocus, mobil ve web client 'ların aynı backend altyapısını kullanarak çalıştığı, seans verilerinin merkezi olarak yönetildiği ve raporlanabildiği uçtan uca bir çalışma takip sistemi olarak tamamlanmıştır.

İÇİNDEKİLER

ÖZET.....	ii
İÇİNDEKİLER	iii
1. GİRİŞ.....	4
2. PROJENİN GENEL TANIMI.....	5
3. SİSTEM MİMARİSİ.....	6
<u>3.1 Genel Mimari Yapı.....</u>	6
<u>3.2 Mobil ve Web Backend ile İlişkisi</u>	6
4. BACKEND KATMANI	7
<u>4.1 Genel Yapı ve Mimari Yaklaşım.....</u>	7
<u>4.2 Modüler Yapı ve Modüllerin Organizasyonu.....</u>	7
<u>4.3 API Üzerinden Veri İşleme ve Raporlama</u>	8
<u>4.4 Kimlik Doğrulama Süreci.....</u>	8
5. VERİTABANI YAPISI.....	9
<u>5.1 Genel Veritabanı Yapısı</u>	9
<u>5.2 Çalışma Seanslarının Saklanması</u>	9
6. KULLANICI ARAYÜZLERİ	10
<u>6.1 Mobil Uygulama.....</u>	10
<u>6.2 Web Raporlama Paneli</u>	11
7. SONUÇ VE GELECEKTE YAPILMASI PLANLANAN ÇALIŞMALAR.....	12

1. GİRİŞ

Odaklanarak çalışma süresinin düzenli biçimde takip edilmesi, verimli çalışma alışkanlıklarının kazanılmasında önemli bir ihtiyaçtır. Bu proje kapsamında, kullanıcıların çalışma seanslarını başlatıp durdurıldığı, seans sürelerinin kayıt altına alındığı ve bu veriler üzerinden raporların üretildiği BeeFocus uygulaması geliştirilmiştir. Uygulamanın temel amacı, çalışma verilerinin merkezi olarak saklanması ve anlamlı raporlar halinde sunulmasıdır.

BeeFocus sistemi; mobil uygulama, backend API ve web tabanlı raporlama arayüzünden oluşmaktadır. Mobil uygulama, çalışma seanslarının oluşturulduğu ana client konumundadır. Backend katmanı, seans verilerinin işlenmesinden ve raporlanmasıdan sorumludur. Web tarafında ise, kullanıcı doğrulaması sonrasında yalnızca raporlama amaçlı bir arayüz sunulmakta ve backend üzerinden elde edilen istatistikler görüntülenmektedir. Bu yapı ile mobil ve web client'ların aynı backend altyapısını kullanması sağlanmıştır.

Bu raporda; öncelikle projenin genel yapısı ve sistem mimarisi açıklanmakta, ardından backend ve veritabanı yapısı ele alınmaktadır. Devamında mobil uygulama ve web raporlama arayüzü tanıtılmakta, son bölümde ise elde edilen sonuçlar ve gelecekte yapılabilecek geliştirmeler değerlendirilmektedir.

2. PROJENİN GENEL TANIMI

BeeFocus, kullanıcıların odaklanarak geçirdikleri çalışma sürelerini takip edebilmesi ve bu veriler üzerinden raporlar elde edebilmesi amacıyla geliştirilmiş bir uygulamadır. Sistem, çalışma seanslarının oluşturulması, bu seanslara ait verilerin saklanması ve elde edilen verilerin istatistiksel olarak raporlanması işlevlerini bütüncül bir yapı içerisinde sunmaktadır.

Uygulamanın mobil tarafı, kullanıcı etkileşiminin yoğun olduğu ana bileşeni oluşturmaktadır. Kullanıcılar mobil uygulama üzerinden çalışma seansı başlatmakta ve sonlandırmaktadır. Her bir seans için başlangıç zamanı, bitiş zamanı, süre ve kategori bilgileri backend katmanına iletilerek kalıcı olarak saklanmaktadır. Mobil uygulama, seans yönetimi ve temel raporlamanın yapıldığı client olarak görev almaktadır.

Web tarafında ise yalnızca raporlama amacıyla kullanılan bir arayüz bulunmaktadır. Web uygulamasına kullanıcı doğrulaması ile giriş yapılmakta ve giriş sonrasında backend üzerinden elde edilen rapor verileri görüntülenmektedir. Web arayüzü, seans verilerini doğrudan üretmemekte; mobil uygulama tarafından oluşturulan verileri kullanarak raporların daha geniş bir ekranda ve özetlenmiş biçimde sunulmasını sağlamaktadır. Bu yapı sayesinde web uygulaması, sistemin raporlama ve analiz bileşeni olarak konumlandırılmıştır.

3. SİSTEM MİMARİSİ

BeeFocus uygulaması, server-client mimarisi esas alınarak geliştirilmiştir. Sistem, mobil uygulama ve web tabanlı raporlama arayüzü olmak üzere iki farklı client'a sahiptir. Her iki client da merkezi bir backend API üzerinden veri alışverişini yapmakta ve backend katmanı aracılığıyla veritabanına erişmektedir.

3.1 Genel Mimari Yapı

Uygulamanın genel mimarisi üç ana katmandan oluşmaktadır: frontend katmanı, backend katmanı ve veritabanı katmanı. İstemci katmanında mobil uygulama ve web raporlama arayüzü yer almaktadır. Backend katmanı, sistemin iş mantığını barındırmakta ve client'lardan gelen istekleri işleyerek uygun yanıtları üretmektedir. Veritabanı katmanı ise çalışma seanslarına ait verilerin kalıcı olarak saklandığı bileşenini oluşturmaktadır.

Bu yapı sayesinde client uygulamalar, doğrudan veritabanı ile iletişime geçmemekte; tüm işlemler backend üzerinden kontrol edilmektedir. Böylece veri tutarlılığı ve merkezi yönetim sağlanmıştır.

3.2 Mobil ve Web Backend ile İlişkisi

Mobil uygulama, kullanıcıların çalışma seanslarını başlattığı ve sonlandırdığı ana client 'dır. Mobil tarafta gerçekleştirilen her işlem, backend API'ye gönderilerek işlenmekte ve veritabanına kaydedilmektedir. Raporlama için gerekli olan tüm veriler bu seans kayıtları üzerinden elde edilmektedir.

Web uygulaması ise yalnızca raporlama amacıyla kullanılmaktadır. Kullanıcı doğrulaması sonrasında web arayüzü, backend API'den rapor verilerini almakta ve bu verileri grafikler ve özet bilgiler halinde sunmaktadır. Web tarafında herhangi bir seans oluşturma işlemi yapılmamakta, yalnızca mevcut veriler okunarak görüntülenmektedir.

4. BACKEND KATMANI

BeeFocus uygulamasının backend katmanı, ASP.NET Core Web API kullanılarak geliştirilmiştir ve sistemin server tarafını oluşturmaktadır. Mobil ve web client'lardan gelen tüm istekler bu katman üzerinden karşılanmaktadır, veri işleme ve raporlama süreçleri merkezi olarak backend içerisinde yürütülmektedir.

4.1 Genel Yapı ve Mimari Yaklaşım

Backend geliştirme sürecinin başlangıcında mimari yapı daha önceki çalışmalarından ötürü mikroservis yaklaşımı dikkate alınarak planlanmıştır. Ancak proje kapsamı, geliştirme süresi ve mevcut kaynaklar değerlendirildiğinde, dağıtık bir mikroservis mimarisinin proje için gereğinden fazla karmaşıklık oluşturacağı görülmüştür. Bu nedenle backend mimarisi, mikroservis mimarisine çoğu yönden benzer bir yaklaşım olan modüler monolit yaklaşımına dönüştürülmüştür.

Modüler monolit yapıda sistem tek bir API host altında çalışmaktadır, ancak farklı iş alanları mantıksal olarak ayrılmış modüller halinde yapılandırılmaktadır. Bu yaklaşım sayesinde hem sorumluluk ayrimı sağlanmış hem de geliştirme ve dağıtım süreçleri sade tutulmuştur.

4.2 Modüler Yapı ve Modüllerin Organizasyonu

Backend içerisinde her bir iş alanı ayrı bir class library olarak tasarlanmıştır. Bu modüller bağımsız servisler olarak çalışmamakta; BeeFocus.API projesi tarafından referanslanarak tek bir ASP.NET Core Web API uygulaması içerisinde birleştirilmektedir. Böylece tüm modüller aynı process altında çalışmakta ve ortak bir API yapısı üzerinden client'lara hizmet vermektedir.

Projede Auth, StudySessions, Subjects, Analytics ve Shared gibi modüller yer almaktadır, her modül yalnızca kendi sorumluluk alanına ait işlevleri içermektedir. Bu yapı, kodun okunabilirliğini ve bakımını kolaylaştırmaktadır.

4.3 API Üzerinden Veri İşleme ve Raporlama

Mobil ve web client'lar, backend ile REST tabanlı API üç noktaları üzerinden iletişim kurmaktadır. API üç noktaları Minimal API yaklaşımı kullanılarak tanımlanmış, bu sayede sade ve okunabilir bir endpoint yapısı elde edilmiştir. Client'lardan gelen istekler backend tarafından karşılanmakta ve gerekli veri okuma veya yazma işlemleri gerçekleştirilmektedir.

Backend içerisinde veri okuma ve veri değiştirme işlemleri bilinçli olarak birbirinden ayrılmıştır. Bu ayrı, CQRS yaklaşımı doğrultusunda uygulanmıştır. Çalışma seanslarının oluşturulması, güncellenmesi ve sonlandırılması gibi işlemler veri değiştiren komutlar (command) olarak ele alınırken; raporlama, istatistiksel hesaplamalar ve listeleme işlemleri yalnızca veri okuma amaçlı sorgular (query) üzerinden gerçekleştirilmiştir.

Bu yapı sayesinde, raporlama işlemleri daha sade ve anlaşılır sorgular üzerinden yürütülmüş, veri yazma süreçleri ile veri okuma süreçleri birbirinden net biçimde ayrılmıştır. Özellikle raporlama odaklı web arayüzü için gerekli olan verilerin backend tarafında düzenli ve kontrollü bir şekilde üretilmesi sağlanmıştır.

4.4 Kimlik Doğrulama Süreci

Backend'de kullanıcı doğrulama işlemleri JWT tabanlı bir yapı ile gerçekleştirilmiştir. Kullanıcı doğrulaması sonrasında oluşturulan token, mobil ve web client'ların sonraki API isteklerinde kullanılmakta ve yetkili erişim bu token üzerinden kontrol edilmektedir. Bu yapı sayesinde backend'e yapılan isteklerde güvenli ve merkezi bir kimlik doğrulama mekanizması sağlanmıştır.

5. VERİTABANI YAPISI

BeeFocus uygulamasında, çalışma seanslarına ve raporlamaya ait verilerin kalıcı olarak saklanabilmesi için ilişkisel bir veritabanı yapısı kullanılmıştır. Veritabanı, backend katmanı tarafından yönetilmekte; mobil ve web client'lardan gelen tüm veri okuma ve yazma işlemleri backend üzerinden gerçekleştirilmektedir.

5.1 Genel Veritabanı Yapısı

Veritabanı tasarıımı, uygulamanın temel işlevi olan çalışma seanslarının takibini merkeze alacak şekilde oluşturulmuştur. Sistem içerisinde kullanıcıların gerçekleştirdiği her çalışma seansı, zaman bilgileri ve ilgili ders bilgileri ile birlikte saklanmaktadır. Raporlama işlemleri sırasında ihtiyaç duyulan tüm veriler bu seans kayıtları üzerinden elde edilmektedir.

Veritabanı yapısı, backend tarafından modüler monolit mimari ile uyumlu olacak şekilde kurgulanmış ve her modül kendi sorumluluk alanına ait veriler üzerinde işlem yapacak biçimde yapılandırılmıştır.

Uygulama içerisinde kullanılan ders bilgileri, kullanıcı tarafından dinamik olarak eklenen veriler yerine sistem başlangıcında seed veri olarak veritabanına yüklenmektedir. Bu yaklaşım ile uygulama içerisinde kullanılacak dersler önceden tanımlanmış olmakta ve raporlama işlemlerinde tutarlı bir yapı sağlanmaktadır.

Seed edilen ders verileri, çalışma seansları ile ilişkilendirilerek her seansın hangi derse ait olduğu bilgisi saklanmaktadır. Böylece raporlama ekranlarında ders bazlı çalışma süreleri ve dağılımlar doğru ve tutarlı bir şekilde üretilebilmektedir.

5.2 Çalışma Seanslarının Saklanması

Çalışma seansları, veritabanında temel veri kaynağını oluşturmaktadır. Her bir seans için başlangıç zamanı, bitiş zamanı ve toplam süre bilgileri saklanmaktadır. Bu kayıtlar, hem mobil uygulamada hem de web raporlama arayüzünde gösterilen istatistiklerin üretilmesinde kullanılmaktadır.

Backend tarafından raporlama işlemleri gerçekleştirilirken, bu seans verileri üzerinden günlük ve haftalık toplamlar, ortalama süreler ve dağılım istatistikleri hesaplanmaktadır. Böylece raporlama işlemleri doğrudan ham seans verileri üzerinden merkezi olarak yapılmaktadır.

6. KULLANICI ARAYÜZLERİ

BeeFocus uygulamasında kullanıcı etkileşimi mobil uygulama ve web tabanlı raporlama paneli olmak üzere iki farklı arayüz üzerinden sağlanmaktadır. Her iki arayüz de backend API ile haberleşmekte, ancak kullanım amaçları ve sistem içerisindeki rolleri birbirinden farklıdır.

6.1 Mobil Uygulama

Mobil uygulama, kullanıcıların çalışma seanslarını yönettiği ana client olarak tasarlanmıştır. Proje geliştirme sürecinin başlangıcında mobil uygulama, sürekli olarak backend'e bağlı çalışacak şekilde planlanmıştır. Ancak uygulamanın kullanım senaryosu değerlendirildiğinde, bu yaklaşımın kullanıcıların çalışma alışkanlıkları ile örtüşmediği görülmüştür. Bu nedenle mobil uygulama mimarisi offline-first yaklaşımına dönüştürülmüştür.

Offline-first yapı kapsamında, çalışma seanslarına ait veriler öncelikli olarak mobil client üzerinde saklanmaktadır. Mobil uygulamada yerel veri yönetimi için ISAR tabanlı bir yerel veritabanı kullanılmıştır. Kullanıcı tarafından oluşturulan seans verileri ISAR veritabanında tutulmakta, uygulama çevrimiçi duruma geçtiğinde bu veriler backend API'ye senkronize edilmektedir. Bu yaklaşım sayesinde uygulama, internet bağlantısı olmadan da kesintisiz şekilde kullanılabilmektedir.

Flutter tarafından uygulama, feature-first mimari yaklaşımı kullanılarak geliştirilmiştir. Bu yapıda uygulama; seans yönetimi, raporlama ve ortak bileşenler gibi işlevsel özellikler bazında modüllere ayrılmıştır. Böylece kod organizasyonu daha okunabilir ve sürdürülebilir bir hale getirilmiştir.

Mobil uygulamanın kullanıcı arayüzü tasarımı, geliştirici tarafından Figma kullanılarak hazırlanmıştır. Tasarım sürecinde kullanıcı deneyimi ön planda tutulmuş, ekran akışları ve görsel bileşenler önceden planlanarak uygulamaya aktarılmıştır. Bu sayede mobil uygulama arayüzü, işlevsel olduğu kadar tatarlı ve kullanıcı dostu bir yapı kazanmıştır.

6.2 Web Raporlama Paneli

Web tarafında geliştirilen arayüz, mobil uygulamadan farklı olarak yalnızca raporlama ve takip amacıyla kullanılmaktadır. Web uygulamasına kullanıcı doğrulaması ile giriş yapılmakta ve giriş sonrasında backend API üzerinden sağlanan rapor verileri görüntülenmektedir.

Web arayüzü, React kullanılarak sade ve anlaşılır bir yapı ile geliştirilmiştir. Web tarafında çalışma seansı oluşturma veya güncelleme gibi işlemler yer almamakta; mobil uygulama tarafından üretilen veriler, raporlar halinde sunulmaktadır. Bu yapı ile web uygulaması, sistemin analiz ve raporlama bileşeni olarak konumlandırılmıştır.

7. SONUÇ VE GELECEKTE YAPILMASI PLANLANAN ÇALIŞMALAR

Bu proje kapsamında, kullanıcıların odaklanarak geçirdikleri çalışma sürelerini takip edebileceği, bu verileri analiz edebileceği ve raporlayabileceği BeeFocus uygulaması geliştirilmiştir. Mobil ve web client'ların aynı backend altyapısını kullanması sağlanmış; mobil tarafta offline-first yaklaşımı ile kullanıcı deneyimi ön planda tutulmuştur. Web tarafında ise raporlama odaklı bir arayüz sunularak sistemin analiz kabiliyeti genişletilmiştir. Proje, belirlenen hedefler doğrultusunda çalışır ve bütüncül bir yapı ile tamamlanmıştır.

İlerleyen aşamalarda BeeFocus uygulaması geliştirilmeye açıktır. Bu kapsamda, uygulamaya sosyallik kazandırmak amacıyla kullanıcıların bir araya gelebileceği grup sistemi eklenebilir. Bu sistem ile kullanıcılar belirli gruplar altında birlikte çalışma sürelerini takip edebilir ve motivasyonlarını artırabilirler.

Ayrıca mevcut ders (subject) yapısı genişletilerek, derslere ek olarak müfredata uygun konu başlıkları tanımlanabilir. Bu sayede kullanıcılar yalnızca ders bazında değil, derslerin alt konuları üzerinden de detaylı çalışma takibi yapabilir ve daha ayrıntılı raporlar elde edebilirler.

Gelecekte yapılabilecek bir diğer geliştirme ise uygulamaya yapay zeka destekli bir çalışma asistanı eklenmesidir. Bu asistan, kullanıcıların çalışma alışkanlıklarını analiz ederek öneriler sunabilir, çalışma planları oluşturmaya yardımcı olabilir ve raporlar üzerinden geri bildirim sağlayabilir. Böylece BeeFocus, yalnızca bir takip uygulaması olmaktan çıkararak kullanıcıya aktif destek sunan bir platform haline gelebilir.

Bu geliştirmeler ile BeeFocus uygulamasının kapsamı genişletilerek daha etkileşimli, kişiselleştirilmiş ve güçlü bir çalışma takip sistemi oluşturulması hedeflenmektedir.