Karar Analizi ve Çözümleme – Geliştirme Ortamı (IDE) Değerlendirilmesi

# Amaç

Bu belgenin amacı, PetVerse projesinde kullanılacak entegre geliştirme ortamının (IDE) belirlenmesine yönelik alternatiflerin analiz edilmesi, proje gereksinimlerine en uygun seçimin gerekçelendirilmesidir.

# Kapsam

PetVerse uygulaması için geliştirme ortamı (IDE) seçiminin detaylarını içermektedir.

# Referanslar

Yoktur.

# Roller ve Sorumluluklar

Çalışma sırasındaki rol ve sorumluluklar aşağıdaki gibidir.

## İç Sorumluluklar

Tablo 1 İç Sorumluluklar

|  |  |
| --- | --- |
| **Ad Soyad** | **Rol** |
| [Tunahan Altıntop](https://confluence.turksat.com.tr/display/~cdyilmaz) | Teknik Danışman / Yazılım Mimari Ekibi |
| [Fatih Bozkoyun](https://confluence.turksat.com.tr/display/~feuluturk) | Teknik Danışman / Yazılım Mimari Ekibi |
| [Yağmur Başoğlu](https://confluence.turksat.com.tr/display/~cdyilmaz) | Yazılım Mimari Stajyeri |
| [Sedanur Ayhan](https://confluence.turksat.com.tr/display/~feuluturk) | Yazılım Mimari Stajyeri |

## Dış Sorumluluklar

Yoktur.

# İş Planı

* 1. **Aktiviteler ve Görevler**

|  |  |
| --- | --- |
| Hafta | Aktivite / Görevler |
| 1. Hafta | Mikroservis makalelerinin okunması, domain analizi, Docker Compose kurulumu, temel proje yapısı kurulumu |
| 2. Hafta | UserService ve PetService geliştirme, JWT güvenliği, ActivityService & RabbitMQ kurulumu, Swagger dokümantasyonu |
| 3. Hafta | Health check, loglama, Prometheus & Grafana ile gözlemlenebilirlik, Rate limiting, frontend taslağı (opsiyonel) |
| 4. Hafta | CI/CD entegrasyonu (GitHub Actions), Docker imajları oluşturulması, test senaryoları, sunum ve demo hazırlığı |

## 5.2 Zaman Planı

 1 Temmuz 2025 tarihinde başlandı.

 29 Temmuz 2025 tarihinde sonuçlandı.

# Değerlendirme Kriterleri

PetVerse projesinde kullanılacak geliştirme ortamının aşağıdaki kriterleri sağlaması beklenmektedir.

## Çözüm Sağlanması Gereken Kriterler

 Güvenlik

 Yüksek performans (hızlı çalışma ve tepki süresi)

 Veritabanı sorgulama verimliliği

 Uygulama kolaylığı / kullanıcı deneyimi

 Java ile tam uyumluluk

 Geliştirici verimliliği

* Kod tamamlama (autocomplete)
* Debug ve test kolaylığı (JUnit, Testcontainers, vs.)

 Takım içi uyumluluk ve kurumsal standartlarla örtüşme

## Beklenen Kriterler

Tablo 2 Beklenen Kriterler 1-5

|  |  |  |  |
| --- | --- | --- | --- |
| **Kriterler** | **Yağmur**  **Başoğlu** | **Sedanur**  **Ayhan** | **Ort. Ağırlık Derecesi** |
|
| **Hızlı olması** | 5 | 4 | 4.5 |
| **Güvenlik** | 4 | 5 | 4.5 |
| **Veritabanı sorgulamada verimlilik** | 5 | 4 | 4.5 |
| **Uygulama kolaylığı** | 5 | 5 | 5 |
| **Geliştirici Verimliliği** | 5 | 5 | 5 |
| **Takım içi uyumluluk ve kurumsal standartlarla örtüşme** | 4 | 5 | 4.5 |
| **Java ile tam uyumluluk** | 5 | 5 | 5 |

## Alternatif Çözümler

IDE seçimi için değerlendirilen başlıca alternatifler şunlardır:

* **IntelliJ IDEA**  
  Gelişmiş Java desteği, Spring Boot uyumluluğu ve kurumsal standartlarla örtüşmesi sayesinde tercih edilmiştir.
* **Visual Studio Code**  
  Hafif, geniş eklenti desteği olan bir editör olmasına rağmen mikroservis ve Spring projelerinde sınırlı özellikler nedeniyle ikinci planda kalmıştır.
* **Eclipse IDE**  
  Geleneksel ve kararlı bir IDE olmasına rağmen, modern Java özellikleri ve mikroservis desteği açısından IntelliJ’e göre yetersiz bulunmuştur.

# Değerlendirme Yöntemleri

Tablo 3 Değerlendirme Yöntemleri

|  |  |
| --- | --- |
| **Yöntem Adı** | **Tanımı** |
| Kriter Değerlendirmesi | Belirlenen kriterler ekip tarafından tartışılarak değerlendirilmiştir. |
| Katılıma Açık Tartışma | Ekip içinde, token mekanizmalarının avantajları ve dezavantajları değerlendirilerek uygun olanın seçimi yapılmıştır. |

# Alternatiflerin Değerlendirilmesi

**IntelliJ IDEA**

**Avantajları:**

* Java 21 ve Spring Boot 3.x ile tam uyumluluk
* Gelişmiş modül yapısı ile mikroservis projelerine uygunluk
* Debug, test ve autocomplete gibi geliştirici verimliliği araçlarında güçlüdür
* Docker, Swagger, CI/CD gibi araçlarla entegre çalışabilir
* Kurumsal ortamlar ve büyük projelerde sık tercih edilir

**Dezavantajları:**

* Ultimate sürüm ücretlidir (Community yeterli olsa da bazı özellikler sınırlıdır)
* Yeni başlayanlar için biraz ağır arayüz olabilir

**Visual Studio Code**

**Avantajları:**

* Hafif ve hızlı
* Zengin eklenti ekosistemi (Spring Boot için eklentiler mevcut)
* Ücretsiz ve açık kaynak

**Dezavantajları:**

* Mikroservis ve çok modüllü projelerde yapı yönetimi zayıftır
* IntelliJ kadar güçlü refactor / debug araçlarına sahip değildir
* Otomatik build/test yapılandırmaları için ek eklenti ihtiyacı duyulur

**Eclipse IDE**

**Avantajları:**

* Java tabanlı klasik projeler için yıllardır kullanılan oturmuş IDE
* Büyük kurumsal projelerde hâlâ kullanılmakta

**Dezavantajları:**

* Modern Java sürümleriyle uyumluluğu sınırlı olabilir
* Eklenti desteği günümüzde zayıflamış durumda
* IntelliJ’e kıyasla arayüz ve kullanıcı deneyimi geri plandadır

# Beklenen Kriterlerin Değerlendirilmesi

Tablo 4 Beklenen Kriterlerin Değerlendirilmesi

Kısaltmalar **I** = IntelliJ IDEA **V** = Visual Studio Code

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Kriterler** | **Yağmur**  **Başoğlu** | | **Sedanur**  **Ayhan** | | **Toplam** | |
| **I** | **V** | **I** | **V** | **I** | **V** |
| **Hızlı olması** | 5 | 5 | 5 | 5 | 10\*4.5 | 10\*4.5 |
| **Güvenlik** | 5 | 4 | 5 | 5 | 10\*4.5 | 9\*4.5 |
| **Veritabanı sorgulamada verimlilik** | 5 | 3 | 5 | 4 | 10\*4.5 | 7\*4.5 |
| **Uygulama kolaylığı** | **5** | **3** | **5** | **3** | **10\*5** | **6\*5** |
| ***Toplam*** | | | | | ***185*** | ***147*** |

# Çözümler ve Risk Değerlendirme

Alternatif IDE çözümleri için temel risk değerlendirmesi aşağıdaki gibidir:

* Visual Studio Code: Eklentiye çok bağımlı çalışır, kompleks mikroservis yapılarında istikrar sorunu yaşanabilir.
* Eclipse: Modern Java sürümleriyle uyumluluk sorunları, kullanıcı arayüzünün eski olması ve bazı build/test araçlarıyla düşük entegrasyon riski vardır.
* IntelliJ IDEA: Community sürümde bazı kurumsal özellikler sınırlıdır (örneğin Spring Boot görsel araçları); ancak genel olarak düşük risklidir.

# Seçilen Çözüm ve Seçim Açıklaması

Kriter değerlendirmesi sonucunda, PetVerse projesi için en uygun entegre geliştirme ortamı olarak **IntelliJ IDEA** tercih edilmiştir.

Yapılan analiz ve puanlama sonuçlarına göre IntelliJ IDEA; Spring Boot ile tam uyumluluğu, Java 21 desteği, mikroservis modül yapısına yatkınlığı, gelişmiş debug/test araçları, Docker & Swagger entegrasyonu ve yüksek geliştirici verimliliği ile öne çıkmıştır.

Ayrıca, ekip içi uyumluluk ve Yazılım Mimari Direktörlüğü tarafından da tercih edilmesi, IntelliJ IDEA’yı hem teknik hem de kurumsal açıdan en uygun çözüm haline getirmiştir.

# Ekler

Yoktur.