Merhaba,

PHP tecrübem olmasına karşın Symfony Framework ‘ü ilk defa kullandım. Symfony 4 ‘ü kullandım. İlk defa kullandığım için son sürüm ile başlamış oldum. Benim adıma Güzel ve keyifli bir çalışma oldu. 2 gün Projeyi geliştirmek için 2 gün de Symfony tutorial için, toplam 4 günlük bir süreçte ortaya çıkan ürünü sizlerle paylaşıyorum.

Proje de Developer ve TodoList olarak 2 temel Object bulunmakta. TodoList Object’inden extend edilmiş TodoListModel Abstract Object i bulunmakta. TodoListModel Object’inden extend eden TodoListJSONType1 ve TodoListJSONType2 Obcejtleri bulunmakta. Bu Objcetler Providerları temsil etmekte. 2 Ayrı providerdan veri çekileceği için 2 type oluşturuldu. Yeni bir provider olması durumunda yeni bir type eklenmesi gerekiyor. Type Object’lerinin createFromJSON metodu type bazlı olarak JSON ‘ı parse ediyor ve TodoList türünde Objectler create ediyor.

TodoListCover Factory Objectidir. Bu proje de Factory Design Pattern kullanıldı. Factory Object ‘e AddTodoListCommand’ı ile yüklenen provider dan çekilen JSON gönderiliyor. Factory Object hangi type’a ait olduğunu belirliyor ve o type üzerinden işlem yapıyor.

DeveloperController Class’ı İşin dağıtım algoritmasını ve base.html.twig içerisine tüm datanın render edilmesini sağlıyor. Algoritma aşağıdaki şekilde tasarlandı;

Verilen Tablo;

|  |  |  |
| --- | --- | --- |
| Developer | Süre | Zorluk |
| DEV1 | 1h | 1x |
| DEV2 | 1h | 2x |
| DEV3 | 1h | 3x |
| DEV4 | 1h | 4x |
| DEV5 | 1h | 5x |

Yukarıdaki Tabloya ve mail de gönderilen tüm materyallere göre;

* Developerların günde 9 Saat çalıştığı varsayılmıştır.
* İş Dağılımında developer başına düşen işlerin saat bazlı toplam süreleri 9 a bölünerek verilen işlerin toplam gün değeri round edilerek bulunmuştur.
* Developerların eş zamanlı çalıştığı varsayılarak, toplam iş süresi en fazla süren developerın gün değeri tüm işin toplam gün değerini belirlemiştir.
* Yukarıdaki tabloya göre bir developer 1 saatte zorluk alanındaki seviyeye ait işi bitirebildiği varsayılmıştır.
* Yukarıdaki tabloda 1 saatte yapılan işlerin zorluk seviyesine doğru orantılı bir şekilde iş dağılımı yapılmıştır. Örn: 1 saatte 5x seviye de bir işi bitiren developer 1x seviyedeki işi 1/5 saatte bitirmiş olarak varsayılmıştır.
* Verilen materyallere göre estimated\_duration değerinin tam olarak neyi ifade ettiği anlaşılmadığından hesaplamalara katılmamıştır. Zorluk seviyesi 1 olup estimated\_duration ı 12 saat olan da var, 4 saat olan da var. Zorluk seviyesi bir işin süresine göre mi belirlendi yoksa işin yoğunluğuna göre mi belirsiz. Yukarıdaki developer tablosu doğrudan baz alındığında developer ve 1 saat olarak baz alındı.
* İş Dağılımı işin zorluk seviyesi kolaydan (1), zora(5) doğru sıralanarak yapılmıştır. Sırayla DEV1 den başlayarak dağıtım yapılmıştır.
* 1 saatte tamamladığı iş seviyesi düşük olan developerlara daha az iş ataması yapılmıştır. Böylelikle seviyesinden büyük işlerde vakit kaybetmesi engellenmiş olup yerine işi daha hızlı yapabilecek developera verilmiştir.

Provider ile data çekip DB ye eklemek için AddTodoListCommand yapıldı. Console ‘a;

php bin/console app:add-todo-list <https://example_provider_url.com/some-argumants>

yazarak DB ye yükleme işlemini yapabilirsiniz.

DB deki tablolar için proje dizininde task\_db.sql oluşturuldu. Dilerseniz burada, dilerseniz symfony ile consoldan tabloları create edebilirsiniz.