



Kadir Emre OZCAN

## Giriş

Bu Üst Düzey Tasarım (HLD) Belgesinin amacı, bu projenin modelini, teknolojilerini ve mimarisini temsil etmek için mevcut proje tanımına gerekli ayrıntıları eklemektir. Bu belge, modüllerin yüksek seviyede nasıl etkileşime girdiği için bir referans kılavuzu olarak da kullanılabilir.

## Kapsam

HLD belgeleri, veritabanı yapısı, uygulama mimarisi ve uygulama akışı gibi sistemin yapısını temsil eder. Bu HLD, sistemin çoğu kullanıcısı için anlaşılması gereken, hafif teknik terimler kullanır.

## Tanımlar

- LDAP - Basit Dizin Erişimi Protokolü
- Solr - Apache lucene ve nosql veritabanına dayalı bir arama programı.
- MySQL - Bir veritabanı tipi ve yönetim sistemi.
- LinkedIn - İşverenlerin potansiyel çalışanları araştırmak için kullandıkları bir web platformu veya tersi
- REST API - GET, PUT, POST ve DELETE verisine HTTP isteklerini kullanan bir uygulama programı arayüzü.
- Tomcat - Apache Software Foundation'da Jakarta Projesi kapsamında geliştirilen Java Servlet ve JavaServer Pages (jsp) teknolojilerinin ücretsiz, açık kaynaklı bir uygulaması.
- Apache - açık kaynaklı bir Web sunucusu.
- JSP - javaServer Pages, çoğunlukla kullanıcı düzeyinde etkileşimler için kullanılan dildir.
- JDBC - WebApplication ve Mysql veritabanı arasında java tabanlı bir arabirim.
- Java - projenin temel aldığı ve üzerine inşa edildiği programlama dili.
- DAO - Veri Erişim Nesnesi
- SMTP - Basit Posta Aktarım Protokolü

## **HLD belgesi:**

- Tasarım özelliklerini sunun
- Kullanıcı arayüzünü tanımlayın,
- Tasarım özelliklerini ve projenin mimarisini dahil edin.

## **Genem Tanım**

Bu Web Uygulaması, birkaç farklı bileşenden oluşmaktadır. Front-end de html, css, javascript, jQuery, Bootstrap ve JSP kullanılırken. Server Java Spring Framework'u ile ayağa kaldırılmıştır. Tomcat Server üzerinde çalışmaktadır. Database olarak MySql kullanılmakta ve Xampp Server aracılığı ile aktif edilmektedir. Spring MySql e bağlanırken Hibernate ORM sini kullanılmaktadır. Projenin işlevlerinin yerine getirilmesi için ayrıca OpenLdp ve Sorl Serverlar da kurulmuştur. Projenin Güvenliği için Spring Security ve custom filtrelerden faydalanılmıştır.

## **Genel kısıtlamalar**

- Web Uygulaması kullanıcı dostu ve kolay erişilebilir olmalıdır.
- Kullanıcılar uygulamanın işlevlerini minimal etkileşim ile kullanabilmelidir.
- Web sitesi, anlaşılması kolay bir haritaya sahip olmalıdır.
- Tüm proje, Java veya Java tabanlı teknolojiler kullanılarak uygulanmalıdır.
- Arama işlevleri için bir Nosql DB kullanılmalıdır.
- İK kullanıcıları ise LDAP teknolojisi kullanılarak doğrulanmalıdır.

## **Varsayımlar**

Bu proje, OBSS tarafından Yaz Stajı Mücadelesi olarak verilmiştir. Amaç, Linkedln'e benzer bir java tabanlı İK uygulaması oluşturmaktır. Kullanıcılar giriş yapmak için kendi Linkedln hesabını kullanabileceğinden, ek kullanıcı adı / şifre bilgisi saklanmaz, yeni bir kullanıcı oluşturma yöntemi yoktur. İK kullanıcılarının arka uç

veritabanlarına tam erişime sahip olacağı ve uygulama içinde profillerini oluşturmaları veya yönetmeleri gerekmeyeceği varsayılmıştır. LinkedIn, 3. taraf uygulamalarıyla sınırlı miktarda veri paylaştığından, son varsayım, yeni kullanıcıların Becerileri sisteme kaydettirecekleri varsayımıdır.

## **Kullanıcı Seçimleri**

### **Spring**

Spring Framework, J2EE geliştirmeyi kolaylaştırmayı amaçlayan açık kaynaklı bir uygulama çerçevesidir. Struts veya Hibernate gibi tek katmanlı çerçevelerden farklı olarak, Spring tutarlı ve verimli bir şekilde tüm uygulamaları yapılandırmaya yardımcı olmayı, tutarlı bir mimari oluşturmak için en iyi tek katmanlı çerçeveleri bir araya getirmeyi amaçlamaktadır.

J2EE uygulamalarının 1999 / 2000'deki yaygın uygulamasından bu yana, J2EE pratikte niteliksiz bir başarı olmamıştır. İşlem yönetimi gibi çekirdek orta katman kavramlarına bir karşılama standardizasyonu getirmesine rağmen, çoğu - belki de en çok - J2EE uygulamaları aşırı karmaşık, gelişmek için aşırı çaba sarf etmekte ve hayal kırıklığı yaratan bir performans sergilemektedir. Bahar, sadece sunucu tarafı J2EE uygulamaları değil, geniş bir ortam yelpazesinde uygulanabilirken, Bahar için orijinal motivasyon J2EE ortamıydı ve Spring, J2EE uygulamalarında kullanılmak üzere birçok değerli hizmet sunuyor.

Bahar, çok çeşitli Java uygulamalarında kullanılan güçlü bir Java uygulama çerçevesidir. Düz Eski Java Nesneleri (POJO) için kurumsal hizmetler sağlar. Bahar, sadeleştirme ve test edilebilirliği artırma için bağımlılık enjeksiyonunu kullanır.

1. Spring, kontrolörler, JavaBean modelleri ve görünümler arasında çok temiz bir bölüm sağlar.

2. Spring'in MVC'si çok esnektir. Action ve Form nesnelerini somut mirasa zorlayan Struts'un tersine (böylece, Java'daki somut mirasta tek vuruşunuzu alarak), Spring MVC tamamen arayüzlere dayanmaktadır. Ayrıca, Spring MVC çerçevesinin hemen hemen her bölümü kendi arayüzünde takarak yapılandırılabilir. Tabii ki bir uygulama seçeneği olarak kolaylık sınıfları sağlıyoruz.
3. WebWork gibi ilkbahar, denetleyicilerin yanı sıra denetleyicileri de sağlayarak, birçok isteğin işlenmesinde yaygın olan davranışı dışa vurmaya kolaylaştırır.
4. Spring MVC gerçekten agnostiktir. Eğer istemezseniz JSP'yi kullanmaya zorlanmazsınız; Velocity, XSLT veya diğer görünüm teknolojilerini kullanabilirsiniz. Özel bir görünüm mekanizması kullanmak isterseniz - örneğin kendi şablon diliniz - Spring View arayüzünü kolayca entegre edebilirsiniz.
5. Yaylı Denetleyiciler, IoC aracılığıyla diğer nesneler gibi yapılandırılır. Bu onları test etmeyi kolaylaştırır ve Spring tarafından yönetilen diğer nesnelerle güzelce bütünleşir.
6. Spring MVC web katmanları, zorlayıcı beton mirasından kaçınma ve kontrolörlerin dağıtıcı servetine açık bağımlılığından dolayı Struts web katmanlarından daha kolay test edilir.
7. Web katmanı, bir iş nesnesi katmanının üstünde ince bir katman haline gelir. Bu iyi uygulamayı teşvik eder. Struts ve diğer adanmış web çerçeveleri iş nesnelerinizi uygulamasında size aittir; Bahar, uygulamanızın tüm katmanları için entegre bir çerçeve sağlar.
8. ActionForms Yok. Doğrudan etki alanı nesnelere bağla
9. Daha fazla test edilebilir kod (doğrulama, Servlet API'sine bağlı değildir)

10. Struts Kontrolörlerinize bağımlılıklar yükler (bir Struts sınıfını genişletmeleri gerekir), Bahar, bunu genişletmeyi seçebileceğiniz Denetleyici uygulamaları olmasına rağmen bunu yapmak için sizi zorlamaz.

11. Yay, iş katmanına iyi tanımlanmış bir arayüze sahiptir

12. Spring, JSP (Velocity / XSLT / FreeMarker / XL vb.) Dışındaki görüntüleme teknolojileriyle daha iyi entegrasyon sunar.

## **JDBC**

JDBC, Java Veritabanı Bağlantısı anlamına gelir. Java'nın, uygulamanın veritabanına erişmesini sağlayan ücretsiz bir açık kaynak kodlu uygulama programlama arayüzüdür. Geliştiricilerin sorgu oluşturmaya, Yapılandırılmış Sorgu Dili (SQL) kullanarak ilişkisel veritabanına veri güncellemesine olanak tanır.

## **JDBC İş Akışı**

JDBC, Java Veritabanı Bağlantısı anlamına gelir. Java'nın, uygulamaların veritabanlarına erişmesini sağlayan ücretsiz bir açık kaynaklı uygulama programlama arabirimidir. Geliştiricilerin Sorgulanan Sorgu Dili'ni (SQL) kullanarak ilişkisel bir veri tabanına sorgu oluşturmalarını ve verileri güncellemelerini sağlar.

## **Neden Hibernate**

Hazırda bekletme, bir RDBMS'ye nesneleri eşlemek ve ilişkisel bir veritabanında nesne yönelimli programlama kavramlarını uygulamak için tasarlanmış, Java için ücretsiz, açık kaynaklı nesne-ilişkisel bir eşleme kütüphanesidir.

- Bir veritabanı bağlantısı açar
- Bir JDBC sürücüsü kullanarak veritabanına SQL sorguları gönderir
- JDBC sürücüsü veritabanına bağlanır

- Sonuç kümesini almak için sorguları yürütür
- Verileri, sürücü yöneticisi aracılığıyla uygulamaya gönderir
- Sonuçlar döndüğünde, verileri işler
- Son olarak, bağlantı kapalı

## Deployment Diagram

