

PSP Eğitimi için Kullanıcı Dostu bir Süreç Yönetim Aracı Geliştirme Denemesi

Selen ÇALIŞKAN, Ebru ÇETİNKAYA, Kıvanç DİNÇER,
Emre YILMAZ, Hande ÇAKICI

Yazılım Mühendisliği Araştırma Grubu (HUSE)
Bilgisayar Mühendisliği Bölümü, Hacettepe Üniversitesi, Ankara, Türkiye

kivanc.dincer@hacettepe.edu.tr

Özet. *SEI (Software Engineering Institute – Yazılım Mühendisliği Enstitüsü)* tarafından geliştirilen *PSP (Personal Software Process – Bireysel Yazılım Süreci)* yazılım geliştiricilerin yüksek kaliteli ürünler ortaya çıkarmalarını hedeflemektedir. *PSP* kullanan yazılım geliştiriciler bireysel yazılım geliştirme süreçlerinin performansını topladıkları ölçümlere dayalı olarak analiz edebilirler. Buna bağlı olarak kendi geliştirme süreçlerini iyileştirebilir ve geliştirme işleriyle ilgili daha hassas kestirimler yapabilirler. Alıştırmalar yapılırken sürecin gerektirdiği veri toplama ve analiz işlemleri normalde *SEI*'nin sunduğu bir yazılım aracı ile yapılmaktadır. 2013 Sonbahar döneminde Hacettepe Üniversitesi'nde verilen BBM-491 kodlu Bireysel Yazılım Geliştirme Süreci adlı derste bu araç kullanılmış ve 80 öğrencinin katıldığı bir anket çalışması araçla ilgili geri beslemeler toplanmıştır. Bu geri beslemeler ve daha önce geliştirilmiş diğer araçlar incelenmek suretiyle ideal bir araçla ilgili gereksinimler belirlenmiş ve öğrenciler ve eğiticiler için kullanıcı dostu yeni bir araç geliştirilmesi hedeflenmiştir. Bu bildiri de bu süreçle ilgili deneyimlerimiz paylaşılmaktadır.

Anahtar Kelimeler: Yazılım Mühendisliği, Süreç Yönetimi, *PSP (Personal Software Process)*, Kişisel Yazılım Süreci, *PSP* Eğitimi, Süreç Otomasyon/Yönetim Aracı, Yazılım Mühendisliği Eğitimi.

Abstract. *PSP (Personal Software Process)* methodology developed by *SEI (Software Engineering Institute)* aims to help developers to produce high-quality software products. Software developers who use *PSP* can analyze their personal process performance based on the collected individual performance metrics and improve their personal development processes to perfect their work habits and to make more accurate predictions about their development work. A software tool offered by *SEI* is usually used during exercises for data collection and analysis. The students of *BBM-491 Personal Software Process* class have used this tool in their *PSP* exercises during the 2013 Fall Semester at Hacettepe University and 80 of those students participated in a survey to give feedback about various aspects of the tool. We determined to develop a new and user-

friendlier PSP tool for students and instructors based on this feedback and the analysis of features of other PSP tools. Our experience with this development process is shared in this paper.

Keywords: Software Engineering, Process Management, PSP, Personal Software Process, PSP Training, Process Automation/Management Tool, Software Engineering Education.

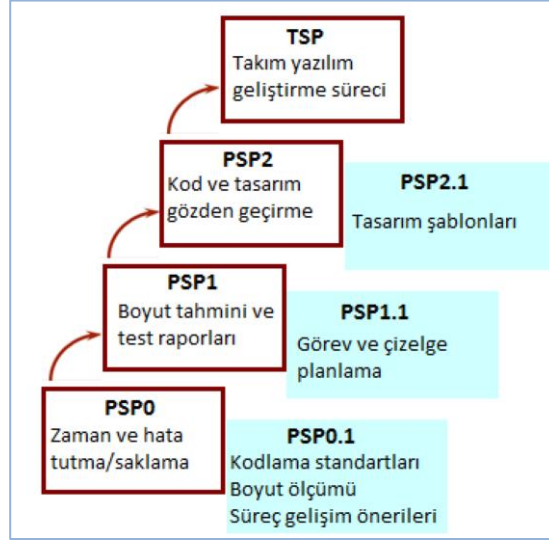
1 Giriş

Yazılım mühendisliğinin tarihçesi 45 seneyi aşmasına rağmen, günümüzde yazılım projelerinin büyük çoğunluğu başarısızlıkla sonuçlanmaktadır. Proje büyüklüğü arttıkça başarı şansı da o oranda azalmaktadır [1]. Humphrey'e göre bunun ana sebeplerinden birisi işi yapanların çoğunun süreç yaklaşımıyla ilgili disipline sahip olmamalarıdır [2]. Sistem büyüdükçe herhangi bir parçada meydana gelen problem tüm sistemi etkilemekte ve problemlerin kaynağının bulunması zorlaşmaktadır, bu yüzden kişilerin çalışmalarının sonuçlarını iyileştiren bir süreç tanımlamak gerekmektedir. PSP tanımlanan bu süreci izleyerek, disiplin içinde yazılım geliştirmeyi sağlayan bir çerçeve sunar. PSP vasıtasıyla endüstrinin büyük projelerinde kullanılan pratiklerin küçük programlama ödevleriyle büyük projelerde kullanmak üzere yazılım geliştiricilere kazandırılması amaçlanmıştır.

Şekil 1'de görüldüğü gibi PSP eğitimi temelde 6 fazdan (PSP0, PSP0.1, PSP1, PSP1.1, PSP2, PSP2.1) oluşmaktadır. PSP eğitimi boyunca geliştiricilerin verilen programlama ödevlerini tek başlarına yapmaları ve bu esnada belli ölçümler yaparak bireysel performanslarını kayıt altına almaları istenmektedir.

Ödevlerde doldurulması zorunlu formlar aşağıda tanımlanmaktadır [5]:

- Zaman Tutma/Saklama Formu: Ödevlerdeki 6 aşama (planlama, tasarım, kodlama, derleme, test etme, kapanış) için harcanan süre kaydı tutulur.
- Hata Tutma/Saklama Formu: Ödev boyunca yapılan hatalar ve düzeltme süreleri tutulur.
- Süreç Geliştirme Önerleri Formu: Detay ve çözüm bilgileri tutulur.
- Test Raporu: Yapılan testlerle ilgili veriler tutulur.
- Büyüklük Tahmini Formu: Önceki ödevlerin kayıtları kullanılarak sonraki ödevlerin kapsamında geliştirilecek ödevlerin kod büyüklüğü tahmin edilir.
- Görev ve Çizelge Planlama Formu: Ödevlerdeki bölümlerin takvime yerleştirilmesi yapılır. Planlanan Değer (*Planned Value - PV*) ve Kazanılmış Değer (*Earned Value - EV*) hesaplamaları burada yapılır.
- Kod ve Tasarım Gözden Geçirme Formu: Kodlama ve tasarım esnasındaki çalışmalar için kontrol listesi (*checklist*) hazırlanır ve çalışmalar gözden geçirilir.



Şekil 1. PSP Süreç Adımları [5]

Bu formların matbu ortamda doldurulmasının ve takibinin zorluğu sebebiyle SEI tarafından formların elektronik olarak doldurulabilmesini sağlayan bir araç sunulmuştur [4]. 2013 Sonbahar döneminde Hacettepe Üniversitesi'nde verilen BBM-491 kodlu Bireysel Yazılım Süreci dersinde SEI'nin sunduğu bu araç kullanılmıştır. Dönem boyunca ve dönem sonunda yapılan değerlendirmelerinde çok sayıda öğrencinin bu araçla ilgili şikayetlerini iletmesi üzerine, dersi alan 80 kadar öğrenciye araçla ilgili bir anket yapılmıştır. Anketten elde edilen geri beslemeler doğrultusunda aracın bazı yapılandırma problemleri olduğu, kullanımının zor olduğu ve kullanım sırasında karşılaşılan hatalardan dolayı etkili bir şekilde kullanılmadığı anlaşılmıştır. Aracın kullanımı sonucunda edinilen tecrübe ile bu konuda benzer diğer araçlar da incelenip [7-9] her birinin iyi yönlerini içeren, eksiklerini gideren, gelişmiş özellikler sunan, PSP eğitimi alan öğrencilere her platformdan erişim ve hatasız bir kullanım olanağı sağlayan, kullanıcı dostu bir araç geliştirilmesi hedeflenmiştir. Bunlara ek olarak, bu yeni araç (HUSE-PSP Tool) vasıtasıyla yazılım geliştiriciler PSP eğitimi sırasında, hem kendi performanslarını diğer yazılım geliştiricilerle karşılaştırabilecekleri hem de süreçlerinin gelişme oranını ve performanslarının yeterlilik düzeyini genel tablo içinde görebilmelerini sağlayacak ek özellikler sunulması hedeflenmiştir.

PSP eğitimini desteklemek için geliştirilen diğer araçlarla ilgili analizlere 2. Bölümde yer verilmektedir. 3. bölümde SEI'nin sunduğu araç ile ilgili öğrencilerden gelen geri beslemeler paylaşılmaktadır. 4. bölümde geliştirilen yeni aracın gereksinim ve tasarımından bahsedilmektedir. Bildiri bu tecrübemizle ilgili önemli hususların ve değerlendirmelerimizin aktarıldığı Sonuçlar bölümüyle son bulmaktadır.

2 PSP Araçlarının Değerlendirilmesi

PSP ile ilgili çalışmalar 1989 yılında başlamıştır. 2006 yılında ise Carnegie Mellon Üniversitesi tarafından *Microsoft Access* ile hazırlanmış bir masaüstü uygulaması olan SEI aracı kullanıma sunulmuştur [6]. Genel hatlarıyla eğitim sürecinde kullanılacak formları içermektedir ve bu formlar aracılığıyla veri kaydının yapılması konusunda kolaylıklar sağlamaktadır. Herhangi bir programlama dili ve tasarım metodu ile kullanılmaya uygundur.

PSP eğitimi için sunulmuş SEI aracının yanı sıra, farklı gruplar tarafından geliştirilmiş birkaç alternatif yazılım aracı daha bulunmaktadır. Derste kullanılan SEI aracı ve incelenen diğer araçlara ilişkin kısa bilgiler ve bu araçların SEI aracına ek olarak tanımlanan özellikleri Tablo 1’de yer almaktadır.

Tablo 1. Mevcut PSP Araçlarının Karşılaştırılması.

Aracın Adı:	SEI [6]		
Platform	İçerdiği Formlar / Ek Özellikler	Mimari	Kısıtlar
Masaüstü	Zaman Tutma/Saklama Formu, Hata Tutma/Saklama Formu, Süreç Geliştirme Önerileri Formu, Test Raporu, Büyüklik Tahmini Formu, Görev ve Çizelge Planlama Formu, Kod ve Tasarım Gözden Geçirme Formu	Form tabanlı bir yapı kullanılmıştır.	MS Access gerekliliği
Aracın Adı:	Dashboard [7]		
Platform	İçerdiği Formlar / Ek Özellikler	Mimari	Kısıtlar
Masaüstü	SEI'nin standart aracındaki formlar bu araçta da mevcuttur. Ek Özellikler: Veri transferine olanak sağlayan <i>import</i> özelliği, Girilen verilerin <i>Excel</i> dosyası olarak saklanmasını sağlayan <i>export</i> özelliği	Proje planlama, tasarım, kodlama, test ve postmortem için scriptler mevcut olup, bu scriptlerde her fazda ne yapılması gerektiği anlatılmaktadır.	Masaüstü uygulaması olması nedeniyle her yerden erişimin sağlanamaması
Aracın Adı	Jasmine [8]		
Platform	İçerdiği Formlar / Ek Özellikler	Mimari	Kısıtlar
Web Tabanlı	SEI'nin standart aracındaki formlar bu araçta da mevcuttur. Ek Özellikler: Manuel veri kaydetme, Veri saklama, Veri analizi,	İki altsistemden oluşmaktadır: * PPMT (Personal Process Management Tool) * PSPG/ER (PSP Guide/Experience)	İstemci tarafında otomatik veri toplamak için kullanılan ve Eclipse'e eklenti olarak çalışan sensörler

	Kazanılmış değer (EV) hesabı Kullanıcı/proje yönetimi	Repository) Client-Server mimarisine göre tasarlanmıştır.	bulunmaktadır. Bu sebeple geliştiricilerin programlama dili olarak Java, geliştirme ortamı olarak ta Eclipse kullanmaları zorunludur.
Aracın Adı	Zelda [9]		
Platform	İçerdiği Formlar / Ek Özellikler	Mimari	Kısıtlar
Masaüstü	SEI'nin standart aracındaki formlar bu araçta da mevcuttur. Ek Özellikler: Proje verisi Gereksinim analizi, Süreç <i>script</i> leri, Üretkenlik veri tabanı, Bakım (<i>maintenance</i>) alt fonksiyonları	Sensör tabanlı	Microsoft Access gerekliliği

Sensör tabanlı olup derleyicilere eklenti olarak çalıştırılan ve otomatik olarak verileri toplayan araçların sürecin takip edilmesi ve analiz edilmesi açısından tutarsızlığa yol açtığı gözlenmiştir. Bu çalışmalar doğrultusunda eklenmesi ve düzeltilmesi öngörülen hususlar yeni geliştirilen araca eklenmiştir.

3 SEI Aracıyla İlgili Geribildirimler

Aracın gereksinimleri belirlenirken mevcut araçların değerlendirmesinden çıkan sonuçlara ek olarak SEI aracını kullanan öğrencilerin geri beslemelerinden istifade edilmiştir. Bölüm 1’de bahsedilen anketten elde edilen sonuçlar [10] analiz edilip yeni geliştirilen ve HUSE PSP Aracı adı verilen araçta, beğenilen özellikler muhafaza edilip beğenilmeyen özellikleri iyileştirme yoluna gidilmiştir. Böylece hem kullanımı kolay, hem herkes tarafından rahatça anlaşılabilen ve herkesin kullanmaktan memnun kalacağı bir bireysel yazılım geliştirme aracı yapılması amaçlanmıştır.

Dersi alan öğrencilere sorulan sorular ve verilen cevaplara ilişkin değerlendirmelere aşağıda yer verilmiştir:

Soru 1. Laboratuvar dersinde kullanılan aracın edinim, kurulum ve yapılandırmasıyla ilgili ne tür problemler yaşadınız?

- Kullanıcıların çoğu 32 bit / 64 bit uyumsuzluğundan bahsetmişlerdir.

- Araç MS Access ile gerçekleştirildiği için program ancak Access'in eski bir sürümüyle çalışabilmektedir.
- Aracın edinim ve dağıtımının bir sorumlu tarafından yapılması gerekmektedir.

Soru 2. Laboratuvar dersinde ilgili aracın kullanım açısından ne tür problemler yaşadınız ve araca ait sevmediğiniz özellikler nelerdir?

Bu soruya verilen cevaplar kullanım zorlukları, eksik fonksiyonlar ve aracın hataları olmak üzere üç alt başlık altında gruplandırılmıştır.

a) Kullanım Zorlukları

- Ara yüzlerin kullanışsız olması, içeriğinin çok karmaşık ve anlaşılmasının zor bir yapıda olmasından şikâyet edilmektedir.
- Platforma bağımlı olmasından dolayı problemler yaşanmaktadır.
- Ödevleri teslim etmek için kullanılan yöntem (*screenshot* alma zorunluluğu) kullanıcıya ek yük getirmektedir.

b) Eksik Fonksiyonlar

- Aracın yapması gereken hesaplamalarda eksikler ve yanlışlar mevcuttur.
- Aracın kullanımı ile ilgili kullanım kılavuzu gibi bilgilendirici bir doküman bulunmamaktadır.

c) Aracın Hata Durumları

- Veriler düzgün olarak kaydedilememekte, bozulabilmektedir.
- Belgelere erişim, kaydetme ve verilerin görüntülenmesi sırasında problemlerle karşılaşmaktadır.
- Programın işleyişi sırasında sebebi anlaşılamayan hatalar meydana gelmekte ve çözümlenemeyen hata mesajları ile karşılaşmaktadır.

Soru 3. Laboratuvarda kullanılan araçla ilgili beğendiğiniz özellikleri sıralayınız.

- PSP eğitiminin temel avantajlarından olan performans ile ilgili bazı istatistiklerin hesaplanıp verilerin düzenlenebilir olarak gösterilmesi, kullanıcıların kendi performansını analiz edebilmesinin sağlanması ve formların hazır olarak kullanıcıya sağlanması gibi beğenilen özelliklerin olduğu saptanmıştır.
- Katılımcıların %5'i, araçla ilgili beğendikleri herhangi bir özellik bulunmadığını belirtmişlerdir.

- Kullanımı kolaylaştırmak amacıyla butonların yanına küçük bilgi kutucukları (hint) eklenebilir.
- Araçla ilgili kullanım kılavuzu eklenebilir.
- Şablonlar ve Scriptler araca bütünleşik hale getirilebilir.
- Yazılım geliştiricilerin kendi performanslarını aracı kullanan diğer kişilerle karşılaştırmasını sağlayacak özellikler eklenebilir.
- Ödevlerin teslimi için kolay bir yöntem geliştirilebilir.

PERSONAL SOFTWARE PROCESS
Hacettepe University

ADMIN INFORMATION
USER : Tuğba GÜRGEN ERDOĞAN
USER TYPE : Admin
USERNAME : pspadmin
LAST LOGIN : 20.11.2011 13:00:00
LOGOUT

Group Name	Group Type	Instructor	User Count		
Hacettepe University	University	Tuğba GÜRGEN ERDOĞAN	3	Update	Delete
Bilkent University	University	Can ALKAN	3	Update	Delete
CS Company	Company	Tufan BEKMEZ	1	Update	Delete

Group Management
Group name:
Note:
Group Type:
Save

Copyright © 2014 Hacettepe University | Design by: Ebru-Emre-Hande-Selen

Şekil 3. Admin kullanıcıya ait grup yönetim ekranı.

4 Fonksiyonlar, Arayüzler ve Tasarım Öğeleri

4.1 Aracın Temel Fonksiyonları ve Arayüzleri

HUSE PSP Aracı'nda üç farklı kullanıcı rolü (Standart User, Instructor ve Admin) bulunmakta olup, bunlara karşılık gelen aktörler UML use case diyagramında tüm aktörlerin desteklemesi gereken gereksinimleri barındıran genel bir User aktöründen kalıtılmıştır. User aktörü sisteme giriş/çıkış, profil bilgisi tanımlama ve güncelleme gibi temel fonksiyonları yerine getirmektedir. Bunun yanı sıra, Instructor (Eğitici); ödev, duyuru, grup yönetimi ve not verme fonksiyonlarını; Standart User, form doldurma, ödev teslim etme, duyuruları takip etme fonksiyonlarını; Admin ise grup ekleme, grup güncelleme, kullanıcı profil bilgilerini görüntüleme gibi genel sistem

fonksiyonlarını gerçekleştirmektedir. Bu aktörler ve ilişkili oldukları kullanım durumları Şekil 2’de görüldüğü gibidir.

SEI aracının kullanımıyla ilgili problemlere çözüm olarak, yazılım geliştiricilerin formların nasıl çalıştığını daha iyi anlamalarını sağlamak amacıyla formların tasarımı gözden geçirilmiş ve basite indirgenerek HUSE PSP Aracı’nda kullanıcı dostu form arayüzleri gerçekleştirilmiştir. Bunlar yapılırken SEI aracının temel form tanımlarında değişikliğe gidilmemiştir.

Yazılım geliştiricilerin çeşitli işlevleri gerçekleştirimi hakkında bilgi almalarını sağlamak amacıyla “*information page*” kısımları eklenmiştir. (Şekil 3 ve 4). Şekil 4’de eğiticinin seçtiği öğrencilerin ödevlerini teslim (*submit*) edip etmediklerinin görüntülediği bir ekran da mevcuttur.

Eklenen bu özelliklerin yanı sıra verilerin kaydedilmesi sırasında yaşanan sorunlar da çözülmüştür. Yazılım geliştiricilerin girdiği veriler düzgün bir şekilde veri tabanına kaydedilip, geliştiricilerin ulaşmak istedikleri geçmiş ödevlere ilişkin verilerine sorunsuz bir şekilde ulaşabilmeleri sağlanmıştır. Öğrencilerin geçmiş ödevlerine ilişkin kaydettikleri veriler daha sonraki ödevlerle ilgili büyüklük ve zaman tahmini yaparken kullanılmakta, ayrıca süreçle ilgili performans metriklerinin hesaplanmasında da yine bu geçmiş verilerden yararlanılmaktadır. Hesaplamalar araç tarafından hatasız ve sorunsuz bir şekilde gerçekleştirilip geliştiricilerin proje takibini ve yönetimini en etkin biçimde gerçekleştirebilmeleri amaçlanmıştır. Kullanılan aracın beğenilen özellikleri geliştirilip iyi ve kullanışlı bir hale getirilerek yeni araca aktarılmıştır.

SEI aracını kullananlardan alınan öneriler doğrultusunda, ödev teslim etme fonksiyonu da geliştirilmiştir. Yeni araçta yazılım geliştiricilerin ödevlerini tamamladıktan sonra “*Gönder*” butonuna basmaları yeterlidir. Geliştiriciler “*Gönder*” butonuna bastıktan sonra gönderilen ödevlerle ilgili, eğiticilerin notlandırma yapabilmeleri sağlanmıştır. Bu özellik sayesinde eğiticilerin kendi sayfalarından istedikleri geliştiricinin formlarını görüntüleyebilmeleri ve isterlerse görüntüledikleri formlarla sorunsuz bir şekilde notlandırma işlemi yapabilmeleri sağlanıp, eğiticilerin geliştiricilerin süreç takibini etkin bir biçimde gerçekleştirmelerine olanak sunulmuştur.

İlk prototipe dahil edilmeyip daha sonraki sürümlerde araca eklenmesi düşünülen özellikler ise aşağıda listelenmektedir:

- Instructor, geliştiricilerin teslim ettiği ödevlere yorum ekleyebilecektir.
- Instructor, kullanıcı ekleme işlemini bir Excel dosyasından verdiği liste ile yapabilecektir.
- Instructor, geliştiricilerin sistem grafiği üzerindeki yerini görebilecektir.
- Instructor, geliştiricilerin sorduğu soruları yanıtlayabilecektir.
- Yazılım geliştiriciler, kendilerini diğer geliştiriciler ile süre, ödev teslim etme zamanı gibi konularda kıyaslayabilecektir.
- Yazılım geliştiriciler, grup arkadaşları ile genel, instructor ile ise özel ya da genel olarak iletişim kurabilecektir.
- Geliştiriciler, duyurulara yorum yapabilecektir.
- Yazılım geliştiriciler şablon oluşturabilecek ve sistemde saklayabilecektir.

- Geliştiriciler, süreç boyunca süreleri, Başlat ve Durdur butonları ile görsel olarak takip edebilecek ve toplam süreyi hesaplayabilecektir.
- Sisteme İngilizce ve Türkçe dil desteği eklenecektir.

PERSONAL SOFTWARE PROCESS
Hacettepe University

HOME
EXPERIMENTS & SUBMITS
GRADE EXPERIMENT
ANNOUNCEMENTS
MANAGE USERS
PROFILE

INSTRUCTOR INFORMATION
USER : Karanç DİNÇER
USER TYPE : Instructor
USERNAME : Id
LAST LOGIN : 20.11.2011 13:00:00
LOGOUT

Users' Submit Information							
User	username	Exp-1	Exp-2	Exp-3	Exp-4	Exp-5	Exp-6
HANDE ÇAKICI	20926209	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
EBRU ÇETINKAYA	20926266	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Copyright © 2014 Hacettepe University | Design by : Ebru-Emre-Hande-Selen

Şekil 4. *Instructor*'a ait öğrenciler ve öğrenciler üzerindeki ödevleri görüntüleme ekranı.

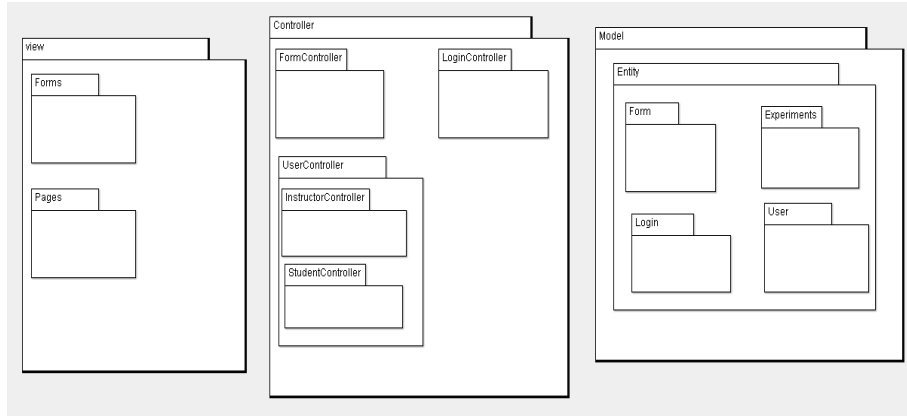
4.2 Mimari Yapı ve Tasarım Öğeleri

Yapılan anketten elde edilen veriler doğrultusunda; kullanılan aracın edinim, kurulum ve yapılandırmasıyla ilgili problemlere çözüm olarak web tabanlı bir bireysel yazılım geliştirme aracı geliştirilmiştir. Böylece platformdan bağımsız olarak yazılım geliştiricilerin, kullanıcı adı ve şifreleriyle kolaylıkla ulaşım sorunsuz kullanabileceği bir araç yapılmıştır.

HUSE PSP aracı .Net platformunda C# programlama dilini kullanarak geliştirilmiştir ve MVC mimari örüntüsü baz alınarak veri ve gösterimin soyutlanması sağlanmıştır. Şekil 5'te sistemin bileşen diyagramı verilmiştir: Sistem üç ana bileşenden oluşmaktadır:

- Model kısmında Form, Experiments, Login ve User olmak üzere dört adet varlık (Entity) sınıfı bulunmaktadır. Verileri saklamak için ise MySQL veritabanı kullanılmıştır.
- View kısmında arayüz tasarımları yer almaktadır, view bileşeni "Forms" ve "Pages" olmak üzere iki gruba ayrılmıştır.
- Controller bileşeni "FormController", "LoginController" ve "UserController" olmak üzere üç gruba ayrılmıştır. Controller sınıfları ile Model ve View katmanları arasındaki veri yönetimi sağlanmaktadır. Böylece veriler (Model) ve kullanıcı ara yüzü (View) birbirini etkilemeden düzenlenebilir [11] Model-View-Controller, bunu Controller adı verilen ara bileşen ile, veri gösterimi ve kullanıcı etkileşiminden, veri erişimi ve iş mantığını çıkarma suretiyle çözmektedir.

Sistemin sınıf diyagramı Şekil.6'da verilmiştir. Bu diyagramlar (use case diyagramı da dahil olmak üzere) tasarım ile ilgili ayrıntı vermektan çok kapsamı genel hatları ile göstermek amacı ile yer verilmiştir.



Şekil 5. Sistemin MVC tasarım örüntüsüne göre yapılandırılan ana bileşenlerini gösteren UML Bileşen Diyagramı.

- [4] W. S. Humphrey, *Introduction to the Personal Software Process*, 1st ed. Addison-Wesley Professional, 1996.
- [5] W. S. Humphrey, *PSP: A Self-Improvement Process for Software Engineers*. Addison-Wesley Professional, 2005.
- [6] W. S. Humphrey, *The Personal Software ProcessSM (PSPSM)*, Addison-Wesley. 2000.
- [7] "The Software Process Dashboard," <http://www.processdash.com/home>
- [8] H. Shin, H.-J. Choi, and J. Baik, "Jasmine: a PSP supporting tool." Springer-Verlag Berlin, Heidelberg ©2007, pp. 73–83, 2007.
- [9] K. Toth, "Simulating Software Product Evolution in the Classroom," 2001.
- [10] K. Dinçer, "BBM-491 Dönem Sonu Ders Değerlendirme Anketi Sonuçları." Aralık 2013.
- [11] "Model-View-Controller," *Vikipedi*, 2014. Available: <https://tr.wikipedia.org/wiki/Model-View-Controller>.