

Hacettepe Üniversitesi Bilgi Gözlem Sistemi	Sürüm: 1
Ödev 4 – Kontrol Listesi + Defect Form	Tarih: 15/05/2015



Ödev 4 – Kontrol Listesi ve Defect Form

Grup 11 – Aurora

Üyeler: Salih Tuç, Melike Bulut, Ebru Alıç, Mert Serin,
Ahmet Nas, Fatıma Betül Tan

Çapraz Grup: Grup 3

15.05.2015

Hacettepe Üniversitesi Bilgi Gözlem Sistemi	Sürüm: 1
Ödev 4 – Kontrol Listesi + Defect Form	Tarih: 15/05/2015

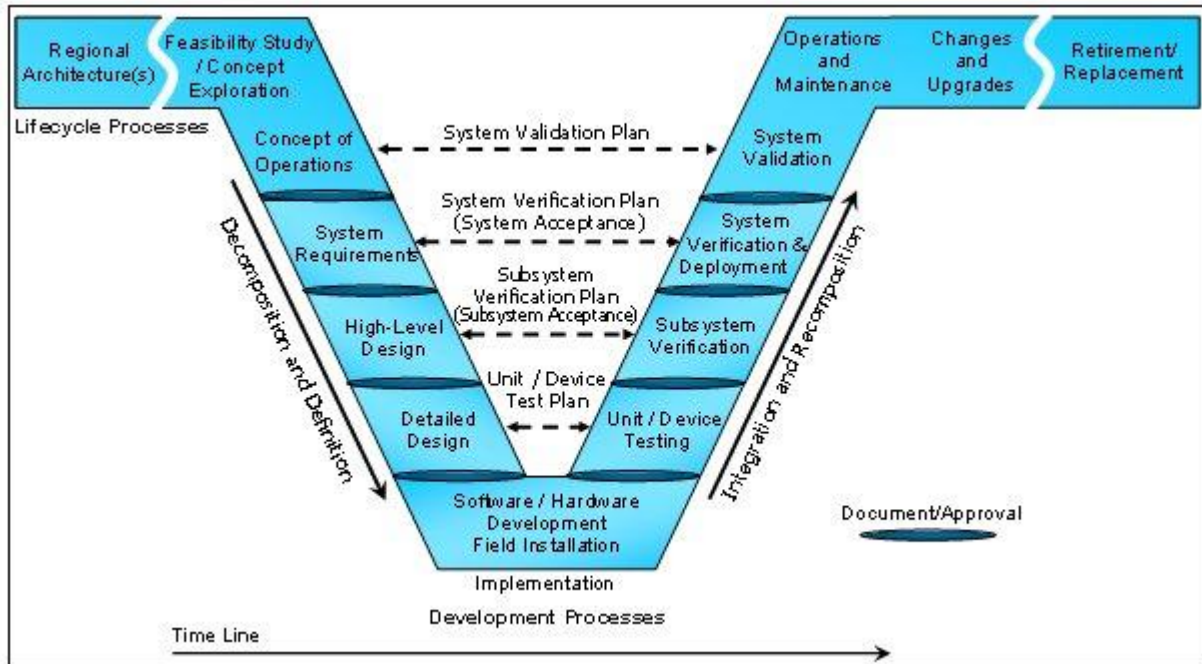
1. Amaç

Bu belgenin amacı, kaynak kitaptan ve standartlardan elde edilen bilgiler yardımıyla bir “Kontrol Listesi” oluşturmak ve bu liste dâhilinde oluşturduğumuz proje ile ilgili bir “Defect Form” çıkartmaktır. Belirlediğimiz kontrol listesi, hem kendi projemizde, hem de çapraz grubumuz olan “3. Grup”un Gereksinim Belgesi’nde denenmiştir. Bunun sebebi, oluşturulan kontrol listesinin ne kadar geçerli veya proje bazlı düşünürsek ne kadar “evrensel” olduğunu göstermektir.

2. Kontrol Listesi

Kontrol Listesi, “Gözden Geçirme (Review)” aşamasının oluşması için kullanılan, statik bir test birimidir. Test edilecek belgede kontrol edilecek durumlar bir liste olarak tutulur. Belgenin bu durumları ele alıp almadığı test edilir ve durum “Defect Form”a işlenir; temel olarak statik test bu mantığa dayanmaktadır.

Statik test kısmı için V Model Test Diyagramı’nın sol kısmını örnek verebiliriz. (bkz. Şekil 1)



Şekil 1: V Model Diyagram

Elde ettiğimiz Kontrol Listesi şu şekildedir:

Hacettepe Üniversitesi Bilgi Gözlem Sistemi	Sürüm: 1
Ödev 4 – Kontrol Listesi + Defect Form	Tarih: 15/05/2015

Kontrol Listesi:

1. Kullanıcı gereksinimleri
2. Sistem gereksinimleri
3. IEEE standartına uygunluk
4. Sürüm geçmişi
5. Yeni sürüm oluşturma nedeni
6. Yeni sürümdeki değişikliklerin özeti
7. Sisteme neden ihtiyaç duyulduğu
8. Teknik terimlerin tanımı
9. User için anlaşılır fonksiyonlar
10. Modüllerin neler olduğu
11. Fonksiyonel ve fonksiyonel olmayan gereksinimler detaylı bir şekilde.
12. Arayüzlerin tanımı
13. Sistem ve bileşenlerin grafik-UML diagram ile gösterimleri
14. Sistemdeki değişiklik(sistem nasıl değişti?)
15. Donanım gereksinimleri (minimal ve optimal bir şekilde)
16. Veritabanı gereksinimleri(veriler ve aralarındaki ilişkiler mantıklı düzenlenmiş olarak)
17. Index
18. Açık bir dil
19. Gereksinim karmaşaları olmaması(func/non-func)
20. Gereksinimlerin birleşmemesi (birlikte ifade edilmemesi)
21. Dilin tutarlı kullanılması
22. Önemli kısımların vurgulanması
23. Bilgisayar terimlerinden kaçınılması
24. Gereksinimlerin neden gerekli olduğu(gerekçesi)
25. Test edilebilir
26. Easy-to-use
27. Gereksinim izlenebilirlik tablosu

Değerlendirdiğimiz gereksinim belgesinin yukarıdaki koşulları sağlayıp sağlamadığına bakılır, beklenen durum, tüm koşulların başarıyla sağlanmasıdır.

Yukarıdaki 18-24. Maddeler arası “doğal dil” kullanıldığı durumda geçerlidir.

Daha sonra da bu maddeler doğrultusunda “Defect Form” oluşturulur.

Submit dosyasında iki adet Defect Form yer almaktadır. Bunlardan “Grup 11 – Aurora - Defect Form” klasöründeki bizim projemizin, “Grup 3 – Defect Form” klasöründeki ise çapraz grubumuzun Defect Form’udur.

Bu belge yakın bir tarihte GitHub ambarına eklenecektir.