**T.C.**

**FIRAT ÜNİVERSİTESİ**

**TEKNOLOJİ FAKÜLTESİ**

**YAZILIM MÜHENDİSLİĞİ BÖLÜMÜ**



**Proje Dokümantasyonu**

**(TEZ KONTROL PROGRAMI)**

**Proje Ekibi**

**175541005 Dilek Eda Gündüz**

**Ocak – 2020**

|  |
| --- |
|  |
| 1. **GİRİŞ** |
| * 1. Projenin Amacı   2. Projenin Kapsamı   3. Tanımlamalar ve Kısaltmalar |
| 1. **PROJE PLANI** |
| * 1. Giriş   2. Projenin Plan Kapsamı   3. Proje Zaman-İş Planı   4. Proje Ekip Yapısı   5. Önerilen Sistemin Teknik Tanımları   6. Kullanılan Özel Geliştirme Araçları ve Ortamları   7. Proje Standartları, Yöntem ve Metodolojiler   8. Kalite Sağlama Planı   9. Konfigürasyon Yönetim Planı   10. Kaynak Yönetim Planı   11. Eğitim Planı   12. Test Planı   13. Bakım Planı   14. Projede Kullanılan Yazılım/Donanım Araçlar |
| 1. **SİSTEM ÇÖZÜMLEME** |
| * 1. **Mevcut Sistem İncelemesi**      1. Örgüt Yapısı      2. İşlevsel Model      3. Veri Modeli      4. Varolan Yazılım/Donanım Kaynakları      5. Varolan Sistemin Değerlendirilmesi   2. **Gereksenen Sistemin Mantıksal Modeli**      1. Giriş      2. İşlevsel Model      3. Genel Bakış      4. Bilgi Sistemleri/Nesneler      5. Veri Modeli      6. Veri Sözlüğü      7. İşlevlerin Sıradüzeni      8. Başarım Gerekleri   3. **Arayüz (Modül) Gerekleri**      1. Yazılım Arayüzü      2. Kullanıcı Arayüzü      3. İletişim Arayüzü      4. Yönetim Arayüzü   4. **Belgeleme Gerekleri**      1. Geliştirme Sürecinin Belgelenmesi      2. Eğitim Belgeleri      3. Kullanıcı El Kitapları |
| 1. **SİSTEM TASARIMI** |
| * 1. **Genel Tasarım Bilgileri**       1. Genel Sistem Tanımı      2. Varsayımlar ve Kısıtlamalar      3. Sistem Mimarisi      4. Dış Arabirimler         1. Kullanıcı Arabirimleri         2. Veri Arabirimleri         3. Diğer Sistemlerle Arabirimler      5. Veri Modeli      6. Testler      7. Performans   2. **Veri Tasarımı**      1. Tablo tanımları      2. Tablo- İlişki Şemaları      3. Veri Tanımları      4. Değer Kümesi Tanımları   3. **Süreç Tasarımı**      1. Genel Tasarım      2. Modüller         1. XXX Modülü            1. İşlev            2. Kullanıcı Arabirimi            3. Modül Tanımı            4. Modül iç Tasarımı         2. YYY Modülü      3. Kullanıcı Profilleri      4. Entegrasyon ve Test Gereksinimleri   4. **Ortak Alt Sistemlerin Tasarımı**      1. Ortak Alt Sistemler      2. Modüller arası Ortak Veriler      3. Ortak Veriler İçin Veri Giriş ve Raporlama Modülleri      4. Güvenlik Altsistemi      5. Veri Dağıtım Altsistemi      6. Yedekleme ve Arşivleme İşlemleri |
| 1. **SİSTEM GERÇEKLEŞTİRİMİ** |
| * 1. Giriş   2. Yazılım Geliştirme Ortamları      1. Programlama Dilleri      2. Veri Tabanı Yönetim Sistemleri         1. VTYS Kullanımının Ek Yararları         2. Veri Modelleri         3. Şemalar         4. VTYS Mimarisi         5. Veritabanı Dilleri ve Arabirimleri         6. Veri Tabanı Sistem Ortamı         7. VTYS'nin Sınıflandırılması         8. Hazır Program Kütüphane Dosyaları         9. CASE Araç ve Ortamları   3. Kodlama Stili      1. Açıklama Satırları      2. Kod Biçimlemesi      3. Anlamlı İsimlendirme      4. Yapısal Programlama Yapıları   4. Program Karmaşıklığı      1. Programın Çizge Biçimine Dönüştürülmesi      2. McCabe Karmaşıklık Ölçütü Hesaplama   5. Olağan Dışı Durum Çözümleme      1. Olağandışı Durum Tanımları      2. Farklı Olağandışı Durum Çözümleme Yaklaşımları   6. Kod Gözden Geçirme      1. Gözden Geçirme Sürecinin Düzenlenmesi      2. Gözden Geçirme Sırasında Kullanılacak Sorular         1. Öbek Arayüzü         2. Giriş Açıklamaları         3. Veri Kullanımı         4. Öbeğin Düzenlenişi         5. Sunuş |
| 1. **DOĞRULAMA VE GEÇERLEME** |
| * 1. Giriş   2. Sınama Kavramları   3. Doğrulama ve Geçerleme Yaşam Döngüsü   4. Sınama Yöntemleri      1. Beyaz Kutu Sınaması      2. Temel Yollar Sınaması   5. Sınama ve Bütünleştirme Stratejileri      1. Yukarıdan Aşağı Sınama ve Bütünleştirme      2. Aşağıdan Yukarıya Sınama ve Bütünleştirme   6. Sınama Planlaması   7. Sınama Belirtimleri   8. Yaşam Döngüsü Boyunca Sınama Etkinlikleri |
| 1. **BAKIM** |
| * 1. Giriş   2. Kurulum   3. Yerinde Destek Organizasyonu   4. Yazılım Bakımı      1. Tanım      2. Bakım Süreç Modeli |
| 1. SONUÇ |
| 1. KAYNAKLAR |

|  |
| --- |
|  |
| **1.GİRİŞ** |
| * 1. **Projenin Amacı**   Projenin amacı, yazılan herhangi tezlerin doğru yazılıp yazılmadığını kontrol eden program   * 1. **Projenin Kapsamı**   Proje dökümanı veya hazırlanan proje raporlarını inceler.   * 1. **Kütüphanede Tanımlamalar ve Kısaltmalar**   docx Kütüphanesi:  Microsoft Word (.docx) dosyalarını oluşturmak ve güncellemek için kullanılan bir Python kitaplığıdır.  os Modülü: Ortak bir arayüz üzerinden farklı işletim sistemleri ile tutarlı bir şekilde iletişim kurabilmemizi sağlayacak pek çok fonksiyon ve nitelik sunar. |
| 1. **PROJE PLANI** |
| * 1. **Giriş**   Bu kapsamda Python programlamadaki terminale “python main.py” yazarak ekrana dosya seçme ekranı çıkacaktır. Kullanıcı dosyayı seçtikten sonra raporu görebilecekleridr.   * 1. **Projenin Plan Kapsamı**      * 1. **Proje Zaman-İş Planı**      * 1. **Proje Ekip Yapısı**      * 1. **Kullanılan Özel Geliştirme Araçları ve Ortamları**      * 1. **Proje Standartları, Yöntem ve Metodolojiler**   Spiralin başladığı ilk çeyrek içinde ilk isterler toplanır ve buna göre proje planlaması yapılır. İkinci çeyrekte, ilk tanımlanan isterlere göre risk çözümlemesi yapılır. Üçüncü çeyrekte, risk çözümlemesi sonunda ortaya çıkan isterlerin tanımlanmasındaki belirsizlikleri ortadan kaldırmak için prototipleme yöntemi kullanılır. Gerekirse benzetim(simülasyon) veya diğer modelleme kullanılarak isterlerin daha sağlıklı tanımlanması sağlanır. Dördüncü çeyrekte, müşteri, ortaya çıkan ilk ürünü inceleyerek değerlendirme yapar, önerilerde bulunur. Bu şekilde tanımlanan ilk döngü bir sonraki döngü için bir girdi oluşturur.   * 1. **Kalite Sağlama Planı**     Şekil 1  **1.Ekonomi:** Ekonomik açıdan yazılımın maliyeti her ne kadar ilk seferde pahalı olsa da ileriye dönük düşünüldüğünde ve zaman tasarrufundan ötürü gayet uygundur.  **2.Tamlık:** Projede herhangi bir açık olmamalı ve programda bulunan tüm butonlar textler vs. çalışır ve tamdır.  **3.Yeniden Kullanılabilirlik:** Program her koşulda tekrardan düzenlenip kullanılabilecek.  **4.Etkinlik:** Kullanıcı sistemin her alanına hakim olduğu için sistemi etkin bir biçimde kullanacak. **5.Bütünlük:** Kullanıcı sistemin tüm kısımlarına hakim olacak ve program bir bütün halinde çalışacaktır.  **6.Güvenilirlik:** Proje gerekli güvenlik önlemlerinin alınması yanı sıra şuan devlet bünyesinde bulunan çok yüksek güvenlik önlemli serverlarda saklanacaktır.  **7.Modülerlik**: Modülerlik otomasyonun her seviyesindeki kişinin ayrı ayrı sayfalardan söz sahibi olmasını sağlar. Örneğin: Yönetim modülü, Giriş Modülü…  **8.Belgeleme:** Bu belgeden de anlaşılacağı üzere tam anlamıyla sistemin özeti olacak bu 23 doküman oluşturulmuştur.  **9.Kullanılabilirlik:** Kullanılabilirlik olarak her seviyedeki insana hitap edeceğinden zor renkler karmaşık sistemlerden kaçınılmıştır.  **10.Temizlik:**  **11.Değiştirilebilirlik:** Proje veri tabanını erişme yetkisi olan ve sistem hakkında bilgisi olan herkes sistemde değişiklik yapabilecek.  **12.Esneklik:** Proje farklı platformlarda ve internet üzerinden çalışacağından gayet esnektir. **13.Genellik:** Proje her üniversitede kullanılabileceğinden geneldir. Ve Türkiye genelinde kullanılacaktır.  **14.Sınanabilirlik:** Projedeki pilot bölge uygulaması sınana bilirliğinin göstergesidir.  **15.Taşınabilirlik:** Sistem internet üzerinden kullanılacağından herhangi bir özel cihaz gerektirmez ve istenilen cihazlarda taşınabilir ve kullanılabilir.  **16.Birlikte Çalışabilirlik:** Bu projedeki en büyük sıkıntı olacak veri girişi şuanda var olan ve her bireyin bilgilerinin saklayan sistemle birleşik ve eş zamanlı çalışmakta.   * 1. **Konfigürasyon Yönetim Planı**   Sistemin ilerde kullanıcının yeni istemlerini karşılayamaması veya sistemin yapısındaki bazı bileşenlerin değişmesi sonucu güncelliğini kaybettiğinde olası konfigürasyon planı hazırlandı.  • Sistemde herhangi bir istenmeyen durum halinde,  Durumları için konfigürasyon yönetim planı oluşturuldu   * 1. **Kaynak Yönetim Planı**   Mevcut bir kaynağımız olmadığından kaynak olarak sadece bu proje dokümantasyonu var.  Kaynak yönetiminde şu hususlar dikkate alınacaktır;  • Kalite beklendiği gibi mi?  • Yeterli hata bulunuyor mu?  • Eğer yeterli sayıda hata bulunmuyorsa veya kalite beklenenden daha iyi görünüyorsa kalite gerçekten daha iyi olabilir.  • Proje elemanları yeni bir eğitim aldı mı?  • Yeni bir araç kullanılıyor mu?  • Yeterli vakit harcanıyor mu?  • İnceleme toplantılarında yeterli sürede vakit harcanıyor mu?   * 1. **Eğitim Planı**   Projeden kazanılacak en önemli olaylardan biride eğitimdir. Kullanılacak dillerin arayüz editör ve programların kullanımında hakim olunamaması halinde bu program başarıyla neticelendirilemez. Bu yüzden projede bazı eğitimler alınması gereklidir. Proje kapsamında alınacak olan eğitimler;   * Python 3.8 eğitimi * Github kullanımı eğitimi     Şekil 1.2   * 1. **Test Planı**   Proje gerekli testi yapılıp kullanıcıya sunulmuştur.   * 1. **Bakım Planı**   Projenin bakım planına gelecek her gün kullanılacak bu sistem tüm değişim ve bazı durumlarda yenilik eklenecek tüm bu sistemsel değişiklikler bakım planında yapılacaktır.   * 1. **Projede Kullanılan Yazılım/Donanım Araçlar**   Python Programlama Dili ve bilgisayar |
| 1. **SİSTEM ÇÖZÜMLEME** |
| * 1. **Mevcut Sistem İncelemesi**   Fırat Üniversitesi Tez Yazım kuralları için tezin doğru olup olmadığını kontrol eden program yapıldı.   * + 1. **Örgüt Yapısı**   Örgüt yapısı kullanıcı ve bilgisayar şeklindedir. Tamamen bireysel kullanılacaktır.   * + 1. **İşlevsel Model**      * 1. **Gereksenen Sistemin Mantıksal Modeli**      1. **Giriş**   Mevcut sistemler incelendiğinde sonuca giden yolda epeyce bir eksikler ve resmi olmayan durumlar söz konusu artık bu sistemi Türkiye hukuk standartlarına uydurmak bize kalıyor.   * + 1. **İşlevsel Model**      * + 1. **Genel Bakış**   Kullanıcı tezlerini kontrol etmek için bu program sayesinde kontrol edebileceklerdir. Bunun yanı sıra olayın akış şekli USE-CASE diyagramında mevcuttur.   * 1. **Arayüz (Modül) Gerekleri**      1. **Yazılım Arayüzü**   Projenin çalışması esnasında böyle bir açık verilmemesine özen gösterildi. Gerekli olan her türlü değişiklik seurce kodları üzerinden yapılıp tekrar derlenecek.   * + 1. **Kullanıcı Arayüzü**   Projede kullanıcının arayüzü tasarlanırken herhangi bir şekilde renkler seçilerek tarafsız rahat büyük puntolu yazılı bir arayüz tasarlanacaktır.       * + 1. **İletişim Arayüzü**   İletişimle ilgili olarak iletişim modülümüzü yazacağız lakin üniversite ile yapılan anlaşmada bu modül konmayacak denirse çünkü bu onların hakkı olası bir kötü durumda muhatap biz olmamamız açısından koyulmayabilir.   * 1. **Belgeleme Gerekleri**      1. **Geliştirme Sürecinin Belgelenmesi**   Geliştirme sürecinde genel olarak belgelendirilmesi hem ileriye dönük hem de şimdiki geliştirme sürecinde projenin tamamlanma yüzdesini nerede kalınıp nerelerde eksikler olduğunu genel hatlarıyla göstermesi amacıyla yapıldı. Bunun yanı sıra projeye yeni dahil olan personellerin olaya hakimiyeti açısından bu yönteme başvuruldu.   * + 1. **Eğitim Belgeleri**   Belgeleri Mevcut bir belgemiz bulunmamaktadır.   * + 1. **Kullanıcı El Kitapları**   Bu kısma projenin en son safhasında kullanıcılara verilecek eğitimlerden pilot uygulamalardan yola çıkılarak hazırlanacak.Yani proje sonunda rahat ve kolay kullanımdan dolayı bir eğitim semineri ve bir kullanım kitapcığı hazırlanacaktır. |
| 1. **SİSTEM TASARIMI** |
| * 1. **Genel Tasarım Bilgileri**       1. **Genel Sistem Tanımı**   Kullanıcı sisteme Word dosyasını eklediğinde program gerekli komut ile incelenip rapor belgesi oluşturulacaktır.   * + 1. **Sistem Mimarisi**   Sistemin mimarisinin akış diyagramı şeklinde verilmesinin temel nedeni sistemin işleyiş mantığının nasıl olduğu ve nasıl bir yol çizileceğinin bilinmesidir. Akış diyagramı sistemin temel mantığı hakkında bize fikir verecektir.    Şekil.4   * 1. **Süreç Tasarımı**      1. **Genel Tasarım**   Öncellikle kullanıcı arayüzü oluşturuldu daha sonra kuralları koda döküldü.   * + 1. **Modüller**        1. **XXX Modülü**           1. **İşlev**   Kullanıcının sisteme müdahale edecebileceği ekrana erişmesi için aşması gereken bir modüldür.   * + - * 1. **Kullanıcı Arabirimi**   Bu projede kullanıcıların işlerini kolaylaştırmak için tasarlandı.   * + 1. **Entegrasyon ve Test Gereksinimleri**   Sistemimizin daha öncede belirttiğimiz gibi devletin mevcut veri tabanlarıyla entegrasyon halinde olması gerekir bu yüzden devlet yetkililerinden bir ekibe gereksinimimiz var. Bunun yanı sıra pilot bölgedeki test süresince 1.derece yazılım yetkilileri orda bulunmalı ve sistem testini gözlemleyip notlar çıkartmalı.   * 1. **Ortak Alt Sistemlerin Tasarımı**      1. **Ortak Alt Sistemler**   Kullanılacak sistem öğrenci, çalışan ve öğretmenlerdir.   * + 1. **Güvenlik Altsistemi**   Yazılım sistemlerinin güvenilirliğe ilişkin nicelikleri, kullanıcıların gereksinimlerini karşılayacak şekilde ortaya koymak ve güvenilirliğin hesaplanmasına yönelik verileri toplama, istatistiksel tahminleme, ölçütlerin tespiti, yazılıma ait mimari özelliklerin belirlenmesi, tasarım, geliştirme ve bunlara yönelik çalışma ortamının belirlenmesi ve modellenmesini kapsamaktadır.  **1.** Model seçme ve düzenlemeye yönelik faaliyetlerin temelinde uygun hedeflerin tespit edilmesi bulunmaktadır.  **2.** Hata ve aksaklıkların analiz edilmesi için uygun verilerin tanımlanması gerekmektedir. Örneğin, arıza veya hataları önemine göre sınıflandırmak, hatalar arası ortalama süreyi bulmak, hata nedenlerini araştırmak, hataları bulmaya yönelik test verilerine karar vermek. **3.** Belirtilen hedeflere yönelik veriler modellenir.  **4.** Geçmişe yönelik verilerin zaman bilgilerini de içerecek şekilde elde edilerek yazılım geliştirme sürecine dâhil etmek.  **5.** Yazılım geliştirme sürecinin modellenmesi, hata ile karşılaşılıp, test sürecine başlamak ve model doğrulama işlemlerine gerçekleştirmek.  **6.** Güvenilirlik tahminleme modelinin seçilmesini sağlamak.  **7.** Güvenilirlik modeli tarafından kullanılacak olan parametrelerini tespit etmek.  **8.** Verilen bir noktayı kullanarak gelecekteki olası hatalar hakkında tahmin yapmak.  **9.** Tahmin edilen hata ve arıza oranları ile gerçekleşen değerleri kıyaslamak.   * + 1. **Yedekleme ve Arşivleme İşlemleri**   Depolanan verilerin, herhangi bir nedenle zarar görmesi, sistemin çalışma süreçlerinde ciddi zararlar oluşturabilir. Yaşanabilecek bir felaket durumu sonrasında, depolanan verilerin geri yüklenememesi, sistemin sağlandığı kullanıcılara veya kurumlara çok ciddi zararlar verebilir. Bu nedenle sistemin çalışma süreçlerine bağlı olarak, yedekleme sistemleri kurulmalı ve yedekleme işlemleri günlük olarak takip edilmelidir. Yedekleme sistemlerinin kurulumu; yedeklenecek veri miktarı, yedekleme sıklığı, yedeklenen verinin zaman içerisinde değişme oranı ve maksimum veri kaybı gibi parametrelere bağlıdır. Sistemin birden fazla sunucusunun eş zamanlı yedekleme işlemini yapabilmesi, işletim sistemlerinin kayıt dosyalarını tam ve eş zamanlı olarak yedekleyebilmesi ve işletim sistemleri üzerinde çalışan veri tabanı uygulamasının yedeklerini sistem kapatılmadan alabilmesi gerekmektedir. |

**Kaynaklar**

* **Youtube kanlları**
* **Python kütüphaneleri**