

T.C.
MANİSA CELAL BAYAR ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

YÜKSEK LİSANS TEZİ
ELEKTRİK - ELEKTRONİK MÜHENDİSLİĞİ ANABİLİM DALI
ELEKTRİK - ELEKTRONİK MÜHENDİSLİĞİ BİLİM DALI

ENDÜSTRİ 4.0 UYUMLU BLE TEKNOLOJİSİ İLE ENDÜSTRİYEL
İŞLETMELERDE IOT TABANLI PERSONEL HARİTALANDIRMA,
TAKİP VE YETKİLENDİRME SİSTEMİ GELİŞTİRİLMESİ

Selahattin Alpcan AĞAYA

Danışman
Prof. Dr. Sezai TAŞKIN



MANİSA-2019

**S. Alpcan
AÇAYA**

**ENDÜSTRİ 4.0 UYUMLU BLE TEKNOLOJİSİ İLE ENDÜSTRİYEL
İŞLETMELERDE İOT TABANLI PERSONEL HARİTALANDIRMA, TAKİP
VE YETKİLENDİRME SİSTEMİ GELİŞTİRİLMESİ**

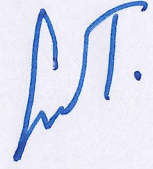
2019

TEZ ONAYI

Selahattin Alpcan AĞAYA tarafından hazırlanan "Endüstri 4.0 Uyumlu BLE Teknolojisi ile Endüstriyel İşletmelerde IoT Tabanlı Personel Haritalandırma, Takip ve Yetkilendirme Sistemi Geliştirilmesi" adlı tez çalışması 27/08/2019 tarihinde aşağıdaki jüri üyeleri önünde Manisa Celal Bayar Üniversitesi Fen Bilimleri Enstitüsü Elektrik - Elektronik Mühendisliği Anabilim Dalı'nda YÜKSEK LİSANS TEZİ olarak savunulmuş ve oybirliği ile başarılı olarak kabul edilmiştir.

Danışman

Prof. Dr. Sezai TAŞKIN
Manisa Celal Bayar Üniversitesi



Jüri Üyesi

Dr. Öğr. Üyesi Hasan SÖZEN
Manisa Celal Bayar Üniversitesi



Jüri Üyesi

Dr. Öğr. Üyesi Özkan AKIN
Ege Üniversitesi



TAAHHÜTNAME

Bu tezin Manisa Celal Bayar Üniversitesi Mühendislik Fakültesi Elektrik - Elektronik Mühendisliği Bölümü'nde, akademik ve etik kurallara uygun olarak yazıldığını ve kullanılan tüm literatür bilgilerinin referans gösterilerek tezde yer aldığını beyan ederim.

Selahattin Alpcan AĞAYA



İÇİNDEKİLER

	Sayfa
İÇİNDEKİLER	I
SİMGELER VE KISALTMALAR DİZİNİ.....	II
ŞEKİLLER DİZİNİ.....	III
TABLO DİZİNİ	IV
TEŞEKKÜR.....	V
ÖZET	VI
ABSTRACT.....	VII
1. GİRİŞ	1
1.1. Literatür Taraması	2
2. GENEL BİLGİLER VE TANIMLAMALAR	8
2.1. Personel Devam Kontrol Sistemi	8
2.1.1. PDKS Avantajları	8
2.1.2. PDKS Zayıf Yönleri	9
2.2. HM-11 Bluetooth Low Energy (BLE) 4.0 Modül.....	9
2.3. Raspberry Pi 3 B+	10
2.3.1. Raspberry Pi 3 B+ Donanım Özellikleri.....	11
2.3.2. İşletim Sistemi	11
2.4. MSSQL Express Server.....	12
2.5. Visual Studio 2017	14
2.6. Python Derleyicisi	15
2.6.1. Libbluet Kütüphanesi.....	17
2.6.2. Pip Kütüphanesi.....	17
2.6.3. Pybluez Kütüphanesi	17
3. MATERYAL VE YÖNTEMLER.....	18
3.1. Tanımlayıcı Etiket Tasarımı	18
3.1.1. HM-11 Modül Konfigürasyonu	18
3.1.2. Bağlantı Şeması	19
3.1.3. Taşıma Kılıfı	19
3.2. Raspberry Pi 3 B+	20
3.3. Algılayıcı Anten	21
3.3.1. Raspberry Pi 3 B+ Konfigürasyonu.....	22
3.3.2. Python.....	22
3.3.3. Yazılım Detayları.....	23
3.4. Visual Studio 2017	24
3.5. Arayüz Tasarımı	25
3.6. Sistem Topolojisi.....	26
3.7. Microsoft SQL Server Express.....	27
3.8. Veritabanı Tasarımı	28
3.9. Web Tabanlı Rapor Arayüz.....	29
4. ARAŞTIRMA BULGULARI VE TARTIŞMA	31
4.1. HM-11 BLE Modül Testleri.....	31
4.2. Yazılım Tasarımı ve Testleri.....	32
4.3. Tanımlayıcı Etiket Tasarımı.....	34
4.4. Saha Testleri	35
5. SONUÇ VE ÖNERİLER	38
KAYNAKLAR	39
ÖZGEÇMİŞ	41
EKLER	

SİMGELER VE KISALTMALAR DİZİNİ

BLE	Bluetooth Low Energy (Bluetooth Düşük Enerji)
BEACON	BLE teknolojisi kullanan işaretçi etiket
RSSI	Sinyal Kalitesi
UUID	Etiket Numarası
PDKS	Personel Devam Kontrol Sistemi
GPS	Küresel Yer Belirleme Sistemi
RFID	Radyo Frekansı ile Tanımlama Teknolojisi
MES	Üretim Yönetim Sistemi
OEE	Toplam Ekipman Etkinliği
SQL	Yapılandırılmış Sorgu Dili

ŞEKİLLER DİZİNİ

	Sayfa
Şekil 1.1. PDKS yazılım arayüzü örneği	1
Şekil 2.1. PDKS giriş-çıkış teminali.	8
Şekil 2.2. HM-11 BLE 4.0 Modül	10
Şekil 2.3. CR2320 Pil ve Pil kutusu	10
Şekil 2.4. Raspberry Pi 3 B+ Genel Görünümü.....	10
Şekil 2.5. Raspberry Pi 3 B+ Temel Özellikleri	11
Şekil 2.6. Raspbian Masaüstü Ortamı	12
Şekil 2.7. MSSQL Örnek Veri Tablosu.	14
Şekil 2.8. Visual Studio C# ile yazılmış bir uygulama.	15
Şekil 2.9. Python dilinde yazılmış bir uygulama	16
Şekil 3.1. Tanımlayıcı etiket kutusu.....	18
Şekil 3.2. Bileklik kordonu	18
Şekil 3.3. Tanımlayıcı etiket tamamlanmış hali.....	19
Şekil 3.4. Raspberry Pi 3 ve Orange Pi karşılaştırma.	20
Şekil 3.5. Raspberry Pi 3 B+ Soğutuculu.....	21
Şekil 3.6. Raspberry Pi 3 B+ muhafaza kutusu.....	22
Şekil 3.7. Python ile mesafe hesaplama formülü.	23
Şekil 3.8. Örnek post cevabı	24
Şekil 3.9. IntelliSense bir örnek.	25
Şekil 3.10. Arayüz ekran görüntüsü	26
Şekil 3.11. Sistem topolojisi	26
Şekil 3.12. MySQL ve MsSQL	27
Şekil 3.13. Web Tabanlı Raporlama Arayüzü Tasarımı Masaüstü Görünümü...	30
Şekil 4.1. Yayın sıklığının pil tüketimi ve konum doğruluğuna etkisi.	31
Şekil 4.2. Arayüz canlı izleme alanı.....	33
Şekil 4.3. Ofis algılayıcı anten konumu.	35
Şekil 4.4. Atölye algılayıcı anten konumu.	35
Şekil 4.5 Web Tabanlı Raporlama Arayüzü Tasarımı Mobil Görünümü.	37

TABLO DİZİNİ

	Sayfa
Tablo 3.1. Sinyaller tablosu örnek veriler	28
Tablo 3.2. Personeller tablosu örnek veriler	28
Tablo 3.3. Canlı izle görünüm örnek veriler	29
Tablo 4.1. Personel Devam Kontrol Tablosu 1 Günlük Örnek.....	36



TEŐEKKÜR

Çalıřmamın her ařamasında bana destek olan, bilgi ve deneyimleri ile yol gösteren danıřmanım Sayın Prof. Dr. Sezai TAŐKIN'a, öğrenim hayatım boyunca beni maddi ve manevi olarak destekleyen ve hep yanımda olan aileme ve eřime yürekten teşekkür ederim.

Selahattin Alpcan AĐAYA
Manisa, 2019



ÖZET

Yüksek Lisans Tezi

Endüstri 4.0 Uyumlu BLE Teknolojisi ile Endüstriyel İşletmelerde IoT Tabanlı Personel Haritalandırma, Takip ve Yetkilendirme Sistemi Geliştirilmesi

Selahattin Alpcan AĞAYA

Manisa Celal Bayar Üniversitesi
Fen Bilimleri Enstitüsü
Elektrik - Elektronik Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Sezai TAŞKIN

Genellikle endüstriyel tesislerde personelin işe giriş çıkış zamanlarının kayıt altına alınması ve bu verilere göre personelin işe devam durumu, fazla mesaisi, vardiya durumu ve maaş hesaplanması gibi raporlama işlemleri için Personel Devam Kontrol Sistemleri kullanılır. Bu sistemin veri giriş cihazları ve personel devam kontrol programı olmak üzere iki ana bileşeni bulunmaktadır. Fakat bu mevcut sistemlerde çalışan personelin tesis içindeki konumu ve iş başında geçirdiği zamanlar otomatik olarak bilinmemektedir.

Bu tez çalışması, tesis içerisinde çalışan personelin belirli alanlarda geçirdiği sürelerin otonom olarak takip edilmesi ile iç mekânda personel haritalandırma, takip ve yetkilendirme sistemi geliştirilmesi üzerinedir. Bu amaçla; BLE modülü olarak HM-11, algılayıcı anten olarak ise Raspberry Pi 3 B+'ta yer alan BLE 4.0 modülü kullanılmıştır. Raspberry Pi B+ dâhili kablosuz ağ bağdaştırıcısı aracılığıyla ağa bağlanılmıştır. Veriler, ASP Classic dilinde yazılan web servislerinin yardımıyla MSSQL Server'a gönderilmiştir.

Arayüz programı, Visual Studio 2017 Geliştirme ortamında C# dilinde yazılmıştır. Veritabanında oluşturulan canlı haritalandırma, SQL yardımıyla personelleri anlık olarak harita üzerinde konumlama yapabilmektedir. Web servisleri yardımıyla raporlar mobil veya kişisel bilgisayarlardan ulaşılabilir bir şekildedir.

Bu çalışmada geliştirilen sistem ile elde edilen raporlar kullanılarak hangi personelin hangi alanda ne kadar süre geçirdiği raporlanarak hızlı bir şekilde takip edilebilmektedir. Geliştirilen yöntem ile tesis içinde çalışan personellerin işbaşındaki çalışma performanslarının daha etkili ve doğru bir biçimde hesaplanabileceği gösterilmiştir.

Anahtar Kelimeler: Beacon, Bluetooth, Personel Takibi, Haritalandırma, BLE, İç Mekân Konum Tespiti, IoT

2019, 40 sayfa

ABSTRACT

M.Sc. Thesis

Development of IoT Based Personnel Mapping, Tracking and Authorization System in Industrial Facilities with Industry 4.0 and BLE Technology

Selahattin Alpcan AĞAYA

**Manisa Celal Bayar University
Graduate School of Applied and Natural Sciences
Department of Electrical and Electronics Engineering**

Supervisor: Prof. Dr. Sezai TASKIN

Personnel Attendance Control Systems are generally used in industrial facilities for recording the starting and finishing of working times of personnel. The recorded data are commonly used for employee attendance status, overtime, shift status and salary calculation. This system has two main components: data acquisition devices and personnel attendance control program. However, the location of the personnels in the facility and the time they spend on the job are not known in current systems.

This thesis focuses on the autonomous monitoring and reporting of the periods spent by the personnel and developing a mapping, tracking and authorization system in the inside of the facilities. For this purpose; HM-11 is used as BLE module and BLE 4.0 module in Raspberry Pi 3 B + is used as sensor antenna. Raspberry Pi 3 B+ is connected to the network through the internal wireless network adapter. Data was sent to MSSQL Server with the help of web services written in ASP Classic.

The interface program is written in C # in the Visual Studio 2017 Development environment. Online mapping created in the database can be used for detecting of personnel location on the map with the help of SQL. With the help of web services, reports can be accessed from mobile or personal computers.

By using the reports obtained with the developed system, it can be followed up rapidly by reporting which personnel spend how much time in which area. With the method, it is shown that the work performances of the personnel working in the facility can be calculated more effectively and accurately.

Keywords: Beacon, Bluetooth, Personnel Tracking, Mapping, BLE, Indoor Location Detection, IoT

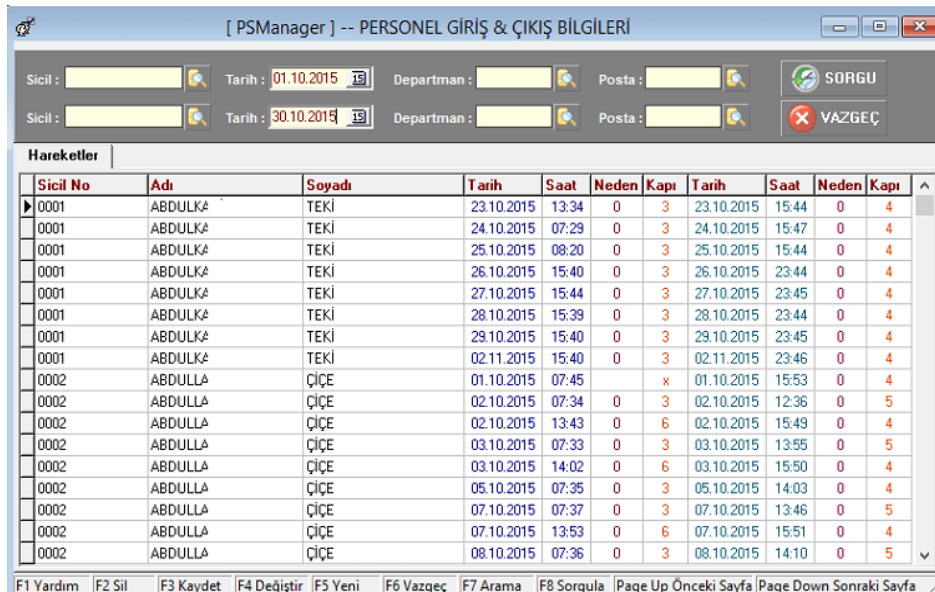
2019, 40 pages

1. GİRİŞ

Endüstriyel tesislerde verimliliği artırmanın unsurlarından birisi personel eğitimi diğeri ise takibidir. Bünyesinde personel çalıştıran bütün işletme ve fabrikalarda, çalışan personellerin sabah ve akşam olmak üzere giriş ve çıkışlarını kontrol etmek amacı ile bilgisayar ortamında kontrol yapılır ve bu bilgiler raporlanarak alınır. Bu sistem işveren ve çalışanlar arasında bir hakem rolü üstlenmektedir. Personel Devam Kontrol Sistemi(PDKS) ile çalışan personel durum kontrolü, kolayca yapılabilmektedir. Bu da işletmelerdeki zaman kaybını ortadan kaldırmaktadır.

PDKS sisteminde kullanılan özel yazılımlar aracılığı ile fazla mesai hesaplaması, aylık maaş ve geriye dönük hesaplamalar yapılabilmektedir. Her gün otomatik olarak tutulan çalışma saatleri ile hesaplamalar kaydedilmekte ve raporlanmaktadır. Bu sistem, personelin işletmeye giriş ve çıkış yaptığı alana kurulmaktadır. Böylece personel takibi ve raporlaması gerçekleştirilmektedir.

PDKS, personelin yalnızca fabrikaya giriş ve çıkışını veya fabrika içerisinde belli noktalardan geçişinin kontrolünü yapar. Bu mevcut sistemde, fabrika içerisinde personel takibi yapılmadığından bu durumu suistimal eden çalışanlar olduğu ve bunun sonucunda da üretimde verim düşüklüğü vb. gibi durumlarla karşılaşıldığı sık sık işletme yetkilileri tarafından dile getirilmektedir.



The screenshot shows the PDKS software interface with the title "[PSManger] -- PERSONEL GİRİŞ & ÇIKIŞ BİLGİLERİ". It features search filters for Sicil, Tarih, Departman, and Posta, along with buttons for "SORGU" and "VAZGEÇ". Below the filters is a table titled "Hareketler" with columns for Sicil No, Adı, Soyadı, Tarih, Saat, Neden, Kapı, and a set of four columns for each date (Tarih, Saat, Neden, Kapı). The table contains 24 rows of data for various personnel and dates.

Sicil No	Adı	Soyadı	Tarih	Saat	Neden	Kapı	Tarih	Saat	Neden	Kapı
0001	ABDULK	TEKI	23.10.2015	13:34	0	3	23.10.2015	15:44	0	4
0001	ABDULK	TEKI	24.10.2015	07:29	0	3	24.10.2015	15:47	0	4
0001	ABDULK	TEKI	25.10.2015	08:20	0	3	25.10.2015	15:44	0	4
0001	ABDULK	TEKI	26.10.2015	15:40	0	3	26.10.2015	23:44	0	4
0001	ABDULK	TEKI	27.10.2015	15:44	0	3	27.10.2015	23:45	0	4
0001	ABDULK	TEKI	28.10.2015	15:39	0	3	28.10.2015	23:44	0	4
0001	ABDULK	TEKI	29.10.2015	15:40	0	3	29.10.2015	23:45	0	4
0001	ABDULK	TEKI	02.11.2015	15:40	0	3	02.11.2015	23:46	0	4
0002	ABDULLA	ÇİÇE	01.10.2015	07:45		x	01.10.2015	15:53	0	4
0002	ABDULLA	ÇİÇE	02.10.2015	07:34	0	3	02.10.2015	12:36	0	5
0002	ABDULLA	ÇİÇE	02.10.2015	13:43	0	6	02.10.2015	15:49	0	4
0002	ABDULLA	ÇİÇE	03.10.2015	07:33	0	3	03.10.2015	13:55	0	5
0002	ABDULLA	ÇİÇE	03.10.2015	14:02	0	6	03.10.2015	15:50	0	4
0002	ABDULLA	ÇİÇE	05.10.2015	07:35	0	3	05.10.2015	14:03	0	4
0002	ABDULLA	ÇİÇE	07.10.2015	07:37	0	3	07.10.2015	13:46	0	5
0002	ABDULLA	ÇİÇE	07.10.2015	13:53	0	6	07.10.2015	15:51	0	4
0002	ABDULLA	ÇİÇE	08.10.2015	07:36	0	3	08.10.2015	14:10	0	5

Şekil 1.1. PDKS yazılım arayüzü örneği

Bu tez çalışmasında geliştirilen sistemde, fabrika içerisinde belirlenen alanlara kurulu olan düşük maliyetli antenler ile personele verilen düşük enerji tüketimli HM-11 Bluetooth Low Energy(BLE) modüllerinin belirlenen alanlar (Üretim alanı, Dinlenme Alanı, Lavabolar, Ofisler v.b. alanlar) içerisinde olup olmadığı, eğer alan içerisinde ise ne kadar süre bu alanda bulunduğuyla ilgili bir analiz yapılmaktadır.

Ayrıca bir sonraki adım olarak yetkin bir personel, vardiya amiri vs. belirlenen üretim alanına giriş yapmadığı sürece makineler çalışmayacaktır. Geliştirilen sistem ile belirli bir personel belirli bir alanda belirlenen süreden fazla kaldığında belirlenen alanda bir alarm sinyali verebilmektedir. Çalışmanın tamamlanarak uygulamaya alınması ile birlikte özellikle fabrikalardaki personelin çalışma performansı daha detaylı raporlanarak performans takibinin daha ayrıntılı bir şekilde yapılabilmesi sağlanacaktır.

Ayrıca, Endüstri 4.0 uygulamalarının başladığı günümüzde bu tez çalışma konusu robotik ve otomasyon sistemlerinin çok daha yaygınlaşacağı bir makineleşme sürecinde de bir çözüm parçası olacaktır. Örneğin, üzerinde BLE modülü taşıyan personelini tanıyan makine ile yetkisiz müdahalelere kapalı bir hale gelerek iş kazalarının da önlenmesi için bir çözüm olacaktır.

1.1. Literatür Taraması

Abdulkareem, A. (2018), BLE beaconlarını kullanarak akıllı bir yoklama takip sisteminin tasarımı, geliştirilmesi ve değerlendirilmesi üzerine çalışmıştır. Yapılan projede öğrencilerin sınıfta olup olmadığını iOS işletim sistemli mobil cihazlar ve BLE beaconları yardımıyla algılayarak akıllı bir yoklama sistemi tasarlanmıştır. Sistemin kararlılığı ve doğruluğunu arttırmak için sınıf içerisinde her noktadan 3 ölçüm alınarak sinyal kaliteleri ve diğer parametreler ölçülmüştür. Sınıf duvarlarına yakın olan öğrencilerin yoklama esnasında doğruluğunu arttırmak için duvar kenarlarında ölçüm yoğunluğu artırılarak her yarım metrede bir ölçülmüştür. Ayrıca dış koridordaki öğrenciler içinde yoklama esnasında hatayı minimuma indirmek için yoğun bir ölçüm gerçekleştirilmiştir. Ölçülen veriler doğrultusunda en ideal yoklama için sınıfta 4 adet beacon cihazı kullanılması gerektiği ortaya çıkmıştır [1].

Aydalka, S.C. (2018), GPS sisteminin kapalı alanlarda hassasiyetini kaybetmesi üzerine beaconlar ile müze ve mağazalarda sesli yönlendirme ve rehberlik uygulamaları üzerine çalışmıştır. BLE teknolojisi kullanarak elektronik devreler ve gömülü yazılımlar aracılığıyla geliştirdiği sistemde kapalı alanlarda lokasyon belirlemeyi hedeflemiştir. Bu sayede müze, mağaza ve sergi vb. yerlerde ziyaretçilere konumlarına bağlı olarak sesli anons ve görsel içeriklerde yönlendirme ve bilgilendirme yapmayı sağlamıştır. Çalışmasında mikrodenetleyici yoluyla BLE sinyallerini işleyerek konum belirleme işlemini gerçekleştirmiştir. Beaconun kimlik numarasıyla birlikte sinyal gücünü de ölçtüğünde en yakın beacon'u bularak kapalı ortam lokasyon bulma işlemini gerçekleştirmiştir. Çalışmasının asıl amacı mağazacılık gibi sektörlerde ziyaretçilere fiziksel olarak buldukları bölümler ile ilgili sesli ve görsel bildirimler göndermeyi ve bu sistemi daha büyük bir entegre tesise uygulanabileceğini göstermiştir [2].

Masmanacı, M.N. (2018), Açık alanlar için geliştirilen konumlama teknolojileri hava, kara ve deniz navigasyon sistemleri, turizm ve otelcilik gibi birçok alanda kullanılmaktadır. Ancak kapalı alanlarda GPS sinyal zayıflaması fazla olduğundan konum tespiti yapılamamakta ve alternatif bir kapalı alan konumlandırma sistemleri geliştirilmektedir. Kapalı alan konum tespit çalışmaları beacon teknolojisinin standartlaştırılmasından sonra daha da ilgi çekici hale gelmiştir. Bu alanda birçok algoritma geliştirilmiş ve her algoritmanın ihtiyaçlara yönelik hesaplama yaptığı görülmüştür. Geliştirilen sistem Bluetooth 4.0 BLE modüller ile beacon mantığı ile çalışmaktadır. Mobil cihazlardan alınan sinyal gücü RSSI değerleri okunup karşılaştırma yapılarak tahmini konum tespiti yapılır. Bulunan sonuçlar Trilateration, Tulip, Cramer rao ve Ağırlaştırılmış ortalama metotlarının sonuçlarıyla karşılaştırılmıştır [3].

Bayılmış, C. ve ark. (2016), Bluetooth Düşük Enerji teknolojisi kullanımı düşük enerji tüketimi, uyumluluk ve fiyat avantajıyla hızla artmaktadır. Genellikle mobil cihazlar tarafından algılanan işaretçi sinyali mobil cihazlar üzerinde konum tabanlı uygulamaları başlatarak etkileşime geçmektedir. Bu özellikler sayesinde konum tabanlı uygulamalar geliştirilebilir. Bu projenin amacı öğrenci yoklama ya da toplantı salonları gibi yerlerde katılım listesi gibi işlemlerin otomatikleştirilmesinde kullanılan RFID, Mobil Cihaz teknolojilerine alternatif olarak BLE teknolojisinin

kullanımını önermektedir. Bu amaçla BLE teknolojisi ve Mobil cihazlar aracılığıyla yoklama takip sisteminin otomatik olarak sunulmaktadır [4].

İlkuçar, M. (2016), GPS Uydu sistemleri ile konum belirleme günümüzde mobil cihazlar v.b. birçok alanda kullanılmaktadır. GPS sisteminde coğrafi konum belirleme GPS uyduları sayesinde elde edilir. Ancak GPS Uydu sinyalleri açık alanlarda çalışmaya uygundur. Kapalı alanlarda sinyal kayıpları çok artacağı için konum belirlemek imkânsız bir hale gelmektedir. Kapalı alanlarda GPS benzeri coğrafi konum belirlemek için farklı teknolojiler geliştirilmiştir. Bunlardan birisi de BLE teknolojisi ile çalışan beacon mantığıdır, belirli mesafeye kadar ayırt edici bilgiler yollayan bu iç mekân vericileri yaydıkları sinyallerin kalitesi ölçülerek vericilere olan uzaklık oranı bulunur. Yapılan çalışmada beacon teknolojisi kullanılarak öğretim üyelerinin odalarına yaklaşan kişiler için ders programına göre uygunluğu, odada olup olmadığı ve müsaitlik durumu gibi bilgileri mobil cihaz üzerinden alınmıştır. Öğretim üyeleri kendi mobil cihazları üzerinden durum güncellemesi yapabilmektedir. Projede Android mobil cihaz, Veritabanı, Web servisleri ve iBeacon kullanılmıştır [5].

Küçük, K. ve ark. (2018), Günümüzde, IoT tabanlı çözümler hızla artmaktadır. Bunun başlıca sebepleri maliyet ve kolay uygulama avantajı sağlamasıdır. Akıllı kentlere doğru yönelirken bu teknolojilerin de dikkate alınması gerekmektedir. Bu çalışmanın amacı, servis araçlarının ve öğrencilerin kampüs içindeki hareketlerinin gerçek zamanlı olarak incelenmesidir. Böylelikle servise biniş ve iniş, okula varıncaya kadarki süre gibi değişkenlerin tüm grupların ulaşabileceği şekilde yayınlanması ve öğrencilerin güvenli bir şekilde okula ulaştırılmasını sağlamaktır. Bu amaç ile HM-11 BLE düşük enerji modülü ile gerçek zamanlı çalışabilen firebase bulut platformunu işaretçi cihazlar ile bağlantı için MIT App Inventor Android mobil uygulama geliştirme platformu ile yazılmış uygulamayı ve web uygulamasını kullanan öğrenci servisi ve öğrenci takibi sunmaktadır [6].

Lin, X. ve ark. (2015), Hastanelerin acil servislerinde hastalar için iç mekân konumlandırma sistemleri üzerine çalışmıştır. Araştırmanın amacı BLE modüller ile Beacon teknolojisine dayanan ve mobil yazılım desteğiyle mobil tabanlı bir iç mekân konumlandırma sistemi uygulamasıdır. Hastaların konumunu bulmak için elde edilen RSSI tabanlı konum tespit sistemi kullanılıyor. Konumlandırma algoritması %97,22

(%95 Güven Aralığı=%95,90-%98,55)sınıflandırma doğruluğunu elde eder. Sonuç olarak mekanizma sağlık personelinin hastaların konumunu takip etme ihtiyacını karşılayabilecek doğruluktadır [7].

Yang, J. ve ark. (2015), Hastanelerde hastaların bölüm veya servislerini bulmaları hastanelerin hızlı bir şekilde çözmesi gereken bir konudur. Bu projede hastaneler için beacon tabanlı iç mekân navigasyon sistemi sunar. Öncelikle beacon'un diğer teknolojilere göre avantajını ele alınmıştır. Devamında hastaneler için iç mekân yönlendirme ve bildirim sistemi tasarlanmıştır. Hastalar SMS göndermek için internet bağlantısı kullanılmıştır. Sonuç olarak hastaneler için iç mekân konumlandırma sistem tarafından başarıyla gerçekleştirilmiştir [8].

Arsan, T. (2018), Düşük enerjili BLE modülleri iç mekân konum bulma sistemlerinde gösterdikleri başarıdan dolayı yüksek oranda kullanılmaya başlanmıştır ve gelişmekte olan bir sistemdir. Bu çalışmada ilave algoritmalar ile birlikte BLE işaretçileri ile beacon mantığı konum belirleme sistemi geliştirilmiştir ve konum doğruluğunun artırılması amaçlanmıştır. Bunun için deneysel iç mekan konum bulma sistemlerinden elde edilen konum verilerinin Büyük Patlama, Büyük Çöküş optimizasyon sisteminden gelen veriler Öklid uzaklık eşleştirme ve Kalman filtresi ile artırıldığı kanıtlanmıştır. Araştırmanın sonucunda konum doğruluğunun %26,62 den %75,69'a artırılmıştır [9].

Aldemir, O. ve ark. (2006), Araştırmalarında görme özürlü bir kişinin tek başına hareket edebilmesi için RFID tabanlı iç mekân konumlandırma ve yönlendirme sistemini tasarlamışlardır. 3 farklı frekansta çalışan UHF vericiler ve Aktif RFID etiketler kullanılmıştır. Bilgisayarla haberleşen sistem oluşturulmuştur. Her verici için ayrı ayrı formüllerle hesaplanan uzaklık verileri bu hesaplamalar sonrasında bulunan konum ile yönlendirme çalışmaları yapılacaktır [10].

Ayabakan, T. ve ark. (2018), Çalıştıkları bildiride, çok algılayıcı veri füzyonu yöntemi olan Federe Kalman Süzgecinin iç mekânda konum belirleme problemine uygulamasını incelemişlerdir. GPS ve Seyrüsefer sistemlerinde merkezileşmiş ve merkezi olmayan yapıdaki kalman süzgeci kullanılmaktadır. Son zamanlarda iç mekânda konum belirleme sistemleri büyük ilgi çekmektedir. Bu bildiride merkezi

olmayan yapıdaki Federe Kalman Süzgeci dış ortamdaki doğruluğu dikkate alınarak iç mekân konum bulma sisteminde kullanılmıştır. Denemeler daha önce hesaplanan değerler ile yapılmış olup, elde edilen sonuçların bundan sonraki sistemlerde kullanılabilir şekilde olduğu değerlendirilmiştir [11].

Uyar, C. ve ark. (2017), İç ortamlarda GPS kullanımını azaltmak için iç ortamlar için daha uygun maliyetli ve daha az yanılma payı olan sistemlere yönelmesi gerektiğini vurgulamışlardır. Bu sistemler RFID, BEACON, Wireless Lan, Geo-Magnetism, UWB gibi kablosuz teknolojilerdir. Bu teknolojileri harmanlanarak daha doğru bir sonuç almak için geliştirilebilir. Dengesizlik ve düşük hassasiyet gibi sorunları BLE Wireless Lan ve Geo-Magnetism teknolojilerinin harmanlanarak bir algoritma oluşturulmasıyla dış ortam gürültüsü azaltılarak doğruluk artacaktır. Bu çalışmada hata değeri ortalama 56cm'dir [12].

İnce, G. ve ark. (2016), GPS verisinin kullanılmayacağı kapalı alanlarda alternatif teknolojiler ile yüksek doğrulukta iç mekân konum belirleme sistemleri üzerine çalışmışlardır. İç mekânda yüksek doğrulukla konum algılayabilmek için konum verisini alabileceğiniz aktarım cihazlarına ihtiyacınız olmaktadır. Bu tezde BLE modülleri beacon modunda kullanılacak ve bize gerekli konum verilerini iletecektir. Farklı entegre üreticilerinin modülleri incelenerek maliyet, stabilizite ve özellik bakımından en uygun ürün Texas Instruments firmasının ana çekirdek olarak kullanılmasına karar verilmiştir. Ana işlemciden sonra diğer donanım ve yazılımlar belirterek geliştirmeye uygun bir hale getirilmiştir. Ana işlemciye bağlanacak yan ürünlerin seçimi gerçekleştirilmiştir. Şemaları ve PCB tasarımları ilgili ortamlarda hazırlanmıştır. Donanım protipi üretilerek dizgi makinelerince imalatı yapılmıştır. Gömülü yazılım faaliyetleri başlamış olup gömülü yazılım tamamlanarak ürünün ilk prototipi hazırlanmıştır. Ürün testlerden geçtikten sonra harici bir firma tarafından tasarlanan kalıplar aracılığıyla dış kutusunun prototipi yapılmıştır ve ürün son testlerinin ardından sahaya çıkabilecek ve rakiplerini zorlayabilecek hale gelmiştir [13].

Özcan, C. ve ark. (2018), Yaptıkları çalışmada üniversitede kullanılmak üzere bir mobil yoklama sistemini BLE modülleri ile yapmışlardır. Üniversitede derse

katılım takibini ve yoklama kâğıdı dolaştırma gibi süreçleri kısaltarak hem öğrenci hemde öğretmenin derse olan ilgisini dağıtmamak ve zamandan tasarruf amaçlanmıştır. Bu çalışmada akıllı mobil cihazlar kullanılarak sınıfa yerleştirilen RFID algılayıcılar ve bluetooth bağlantılı modüller aracılığıyla her öğrenciye önceden verilen bir RFID ile temassız olarak okutulan kartlarda daha önceden karta yüklenen öğrenci numarası alınarak yoklama işlemi gerçekleştirilir. RFID kartları çoğu zaman öğrencinin cebinden çıkarmasına gerek kalmadan kapı önlerine yerleştirilen yüksek güçlü antenler vasıtasıyla öğrenci numaraları alınmaktadır. Böylece ders ve öğrenci ile ilgili istatistiksel veriler hızlı bir şekilde depolanarak incelemeye açık halde gelmektedir. RFID okuyucular belli bir hafıza kapasitesine sahip olup sunucuya aktarım dönem sonunda yapılmaktadır [14].

Chintalapudi, K., ve ark. (2010), Yaptıkları çalışmada kapalı ortam lokasyon bulma üzerine wireless lan vericileri kullanarak bir sinyal haritası oluşturmuşlardır. Sinyal haritası oluştururken kullandıkları mobil cihazlar GPS ile konumlanmanın izin verdiği konumlarda doğrulama amaçlı gps verisini de kullanan bir algoritma ile çıkarttıkları harita ile wireless lan verici cihazlarla olan bağlantı kalitesinden yola çıkarak yapılan sinyal kalitelerini eşleştirerek tahmin ettikleri konumu iç mekân konumlandırma olarak kullanmışlardır. İki farklı binada yaptıkları testler sonucunda çalışmalarının umut verici olduğu ortaya çıkmış. Çalışma öncesi herhangi bir kalibrasyon yapılmamasına rağmen küçük bir binada ve en iyi performans ile elde edilen verilere göre 0.7m ile 4m arasında daha büyük bir binada ise 2m ile 7m arasında konum tahmini yapmışlardır [15].

Ladd, A.M., ve ark. (2004), Yaptıkları çalışmada IEEE 802.11b kablosuz ağ altyapısını kullanarak iç mekânlarda konum tespiti yapılabileceğini göstermiştir. Projelerinde zaten hali hazırda kullanılan bir kablosuz ağın standart olarak sağladığı sinyal kalitesi verisini işleyerek iç mekânlarda konum tespiti yapılmış olup tek dezavantajı hali hazırda iç mekânlarda konum tespiti yapılamayacak sayıda kablosuz ağ dağıtıcısı kullanımınıdır. Daha fazla kablosuz ağ verisi ile yapılan testlerde kablosuz ağ vericisi sayısı arttıkça konum hassasiyeti dahada çok artmaktadır. Okunan sinyal kalitelerinin bayes teoremiyle incelenmesi sonucunda konum algılaması ve canlı takibin yapılabileceğini göstermişlerdir. Bu projede sinyal kalitesi algılaması için mobil cihazlar kullanılmıştır [16].

2. GENEL BİLGİLER VE TANIMLAMALAR

2.1. Personel Devam Kontrol Sistemi

Personel Devam Kontrol Sistemi(PDKS); tesislerde, şantiyelerde, işletmelerde ve fabrikalarda firmaların çalışma sistemlerine göre personelin mesai, fazla mesai, eksik çalışma gibi puantaj haklarını hesaplayan otomasyon sistemidir. PDKS sistemlerinin ana organı yazılımdır. PDKS elemanları ise personel devam takip sistemi cihazları yani giriş-çıkış terminalleridir. PDKS sosyal tesislerden fabrikalara, otellerden hastanelere geniş bir sektöre hizmet eder. Şekil 2.1’de giriş-çıkış terminaline ilişkin bir resim verilmiştir.



Şekil 2.1. PDKS giriş-çıkış terminali

2.1.1 PDKS Avantajları

PDKS'nin bir tesise avantajları sırasıyla; personellerin devam kontrolleri ve yoklamalarının hızlı bir şekilde yapılarak zamandan tasarruf sağlanması, anlık tesis içerisindeki insan sayısının sorgulanabilmesi, tesiste çalışanların çalışma saatlerinin detaylı ve otomatik olarak saklanarak raporlanması, ek mesai veya puantajların yazılım aracılığıyla hızlıca hesaplanması, izinsiz girişlerin önlenmesi olarak sıralanabilir.

2.1.2 PDKS Zayıf Yönleri

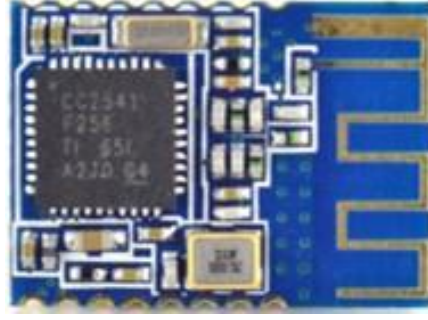
PDKS'nin tesislerde kullanımını esnasında zayıf kalan yönleri; personellerin art niyetli davranarak sistemi aldatabilmeleri, birbirleri yerine kart basma vb. gibi suistimallere açık olması, çalışanların tesis içinde ne kadar etkin çalıştıklarının otonom olarak takibinin kontrol edilememesi sayılabilir. Bu gibi olumsuzluklar günümüz şartlarında personel verimini ve dolayısıyla işletme verimliliğini önemli ölçüde etkilemektedir.

2.2. HM-11 Bluetooth Low Energy (BLE) 4.0 Modül

SMD komponentlere sahip HM-11 BLE modülü TI cc2541 entegresini kullanır. Düşük güç tüketimine sahip olan bu modül fiyatı ve düşük güç tüketimi özellikleriyle ön plana çıkmaktadır. Entegre içerisinde gelen yazılım aracılığıyla AT komutlarını kullanarak parametreleri ayarlanır. Android 4.3 ve üzeri, iPhone ve iPad gibi BLE uyumlu cihazlar ile kullanılabilir. HM-11 modülünün en büyük avantajı beacon modunda çalışabilmesidir.

Beacon modu, BLE modülünün belirlenen aralıklarla uyanıp istenen parametreleri broadcast yayın yapmasını sağlar ve tekrar uykuya geçer. Uyku esnasında derin uykuda olduğu için neredeyse hiç enerji tüketmez ve uyanma süresi sıklığına ve ortam şartlarına göre 6 yıla kadar bir CR2320 pil ile çalıştırılabilir. Beacon çalışma mantığında; etiket numarası, majör ve minör numarası, yayın gücü, mac adresi ve sinyal kalitesi gibi veriler bir BLE 4.0 alıcı modül ile okunabilir.

HM-11 Modülleri farklı üreticiler tarafından üretildiği için katalog değerlerinde literatürde farklı güç tüketimleri yer almaktadır. Fakat tüm modellerde kullanılan CC2541 entegresi aynıdır. Modüldeki 3.3V gerilim regülatörü ve devredeki led vb. komponentlerden kaynaklanan farklı güç tüketimleri oluşmaktadır. Bu çalışmada kullanılan besleme pil gerilimi 3V olduğundan regülatörsüz kullanılmıştır. Dolayısıyla, CC2541 entegresi sinyal gönderirken maksimum 18.6 mA, sinyal alımında en yüksek 18.3 mA, uyku modunda ise 1µA akım tüketimine sahiptir. Gerçekleştirilen işaretçi cihaz sinyal gönderir ve genellikle uyku modlarında çalışır. Bu modlarda tüketilen güç 3V'luk besleme gerilimi kullanıldığı durumda sinyal gönderirken yaklaşık olarak 72 mJ/s, uyku modunda ise 0.0027mJ/s seviyelerindedir.



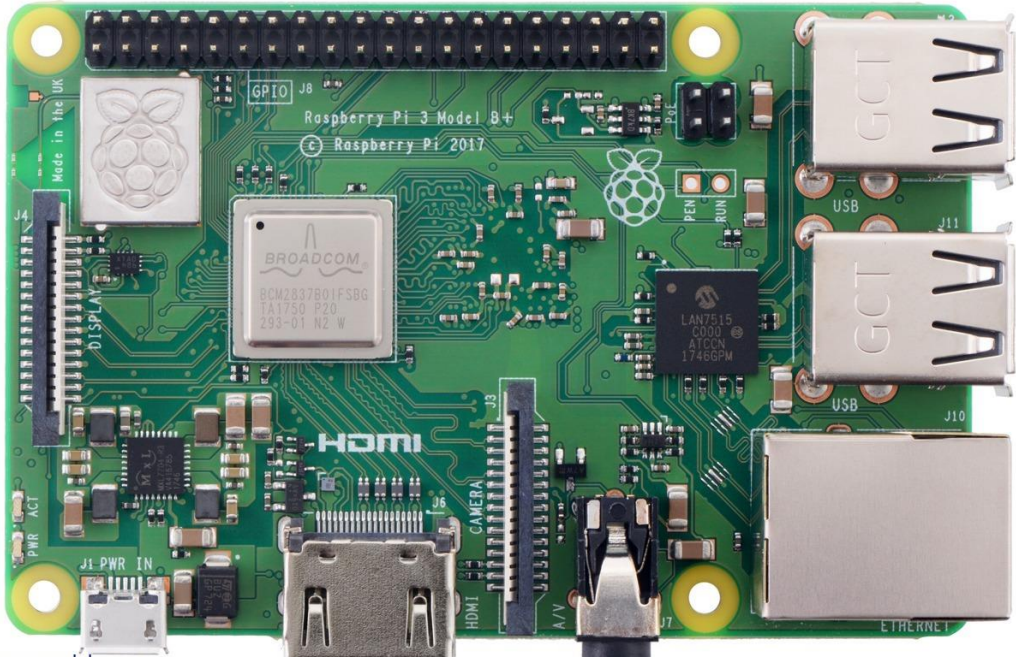
Şekil 2.2. HM-11 BLE 4.0 modül



Şekil 2.3. CR2320 Pil ve pil kutusu

2.3. Raspberry Pi 3 B+

Raspberry Pi bir tek kart bilgisayardır. Bir bilgisayar için gerekli olan ram, işlemci, grafik motoru, giriş ve çıkışların tek bir kart üzerinde toplanmış halidir. Küçük tasarımı ve kompakt yapısı sayesinde büyük alanlara ihtiyaç duymadan orta seviye bir bilgisayar gibi kullanılabilir. Şekil 2.4’de Raspery Pi 3 B+ görülmektedir.

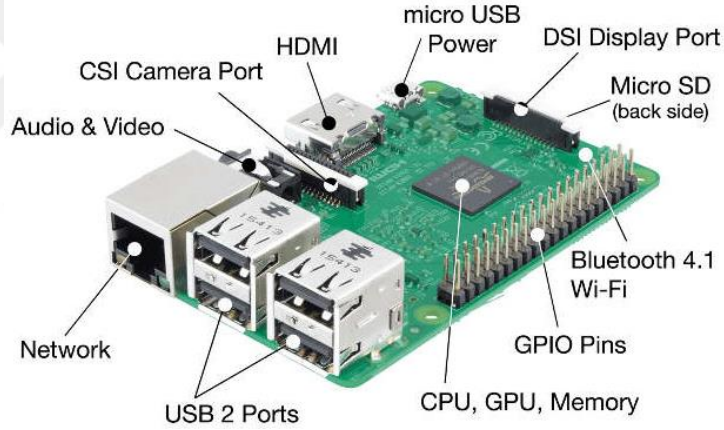


Şekil 2.4. Rasperry Pi 3 B+ genel görünümü

2.3.1. Raspberry Pi 3 B+ Donanım Özellikleri

Raspberry Pi 3 B+'da 1.4GHz hızında çalışan 64-bit destekli ARM Cortex-A53 işlemci bulunmaktadır. Kablosuz bağlantısı da çift band (2.4GHz ve 5GHz destekli) 802.11ac ve Bluetooth desteği 4.1'yi desteklemektedir.

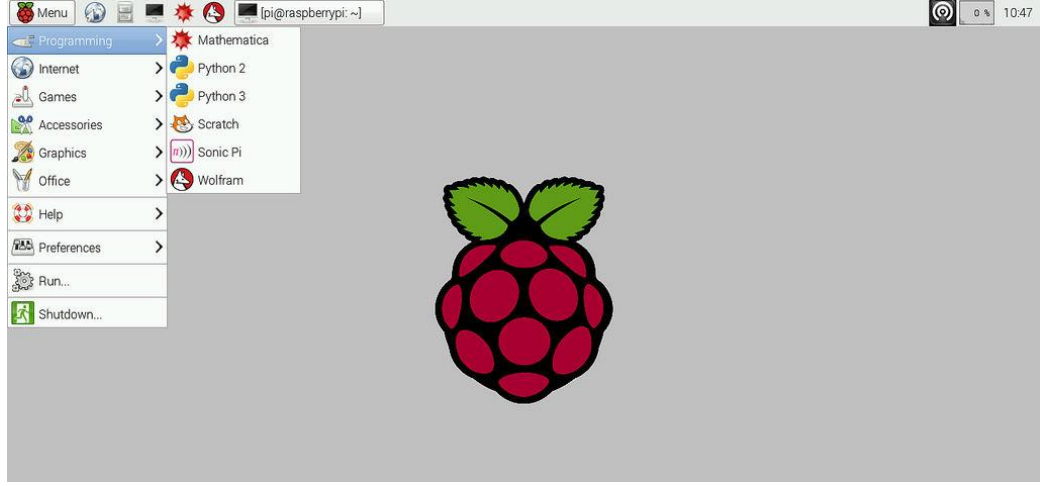
Kablosuz bağlantıların yanı sıra, Raspberry Pi üzerinde bulunan Ethernet bağlantısı PoE, yani Power-over-Ethernet desteği sunmaktadır. RAM olarak 1GB LPDDR2 SDRAM bulundurmakta olup USB 2.0 üzerinden 300Mbps hızında Ethernet modülü desteklemektedir. HDMI üzerinden monitöre bağlanabilir veya DSI Display Port kullanılarak bir LCD ekrana bağlanabilmektedir. Kamera portundan VGA Kamera bağlanabilir. Gücünü Micro USB üzerinden alır. HDMI çıkışı, 4 Adet USB Portu ve ses çıkışı da bulunduran bu mini bilgisayar işletim sistemini bir SD kart üzerinden koşturur.



Şekil 2.5. Raspberry Pi 3 B+ temel özellikleri

2.3.2. İşletim Sistemi

Raspberry Pi 3 B+, işletim sistemi olarak Linux tabanlı Raspbian ile çalışır. Ayrıca Windows 10 IoT core ile de çalışabilir ancak günümüzde genellikle Raspbian işletim sistemiyle kullanılmaktadır. Raspbian, Raspberry Pi için optimize edilmiş Debian'a dayalı ücretsiz bir işletim sistemidir. Bu işletim sistemi, Raspberry Pi'yi çalıştıran temel programlar ve yardımcı programlar kümesidir. Şekil 2.6'da Raspbian masaüstü görsel verilmiştir.



Şekil 2.6. Raspbian masaüstü ortamı

2.4. MSSQL Express Server

MSSQL Express Server, Microsoft firması tarafından geliştirilen ve geliştirilmeye devam eden bir ilişkisel veritabanı sistemidir. Bu sistemlerinde veriler tablolar halinde saklanmaktadır. Kendi aralarında ilişkisel anlamda bağlamlar olabilmektedir.

SQL Server'dan bahsederken bir uygulama olarak bahsetmek yanlış olacaktır. Bir servis olarak çalışır ve SQL Server'ı kurduğunuzda ayarlarda bir değişiklik yapmazsanız Windows ile birlikte SQL Server hizmetleride başlatılır bu hizmetlerin amacı veri kaybını önlemektir. Servisler kısaca Veritabanı aynalama, loglama, yedekleme olarak çalışmaktadır.

Örneğin; bir alışveriş sitesinden alışveriş yaparken ürünlerinizi seçtiğinizde ve siparişi verdiğinizde bilgilerinizin bir yerde saklanması gerekmektedir. Saklanan yer yüksek ihtimalle bir SQL serverdır. Bu siteden alışveriş her an yapılabileceği için 7 gün 24 saat çalışan bir veri tabanı gerekmektedir. Bu gibi durumlarda yardımımıza SQL server koşturmaktadır. SQL serverdan farklı veri depolama sistemleri de mevcuttur ancak SQL server kadar yaygın kullanılmamaktadır.

Bu tip siteler ve uygulamalar için SQL server çok önemlidir. Sürekli veriler üzerinde işlem yapıldığı için verilerin güvenilir ve hızlı bir yapıda saklanması, isteklere cevap vermesi gerekir. Kullanım alanları WEB siteleri, Muhasebe

uygulamaları, Fabrika yönetim uygulamaları v.b. şeklindedir. Bu özelliklerin her biri ayrıca ayrı uzmanlıklar gerektirir. Bir SQL server çok farklı özellikleride bünyesinde barındırır bunlardan bazıları yedekleme yani verilerin bir felaket durumunda kaybedilmemesi için yedeklenmesidir, aynalama yine herhangi bir felaket durumunda sistemin aksamadan devam etmesi ve otomatik olarak işlerin devralınması için için farklı şehir, ülke hatta kıtadaki sunuculara anlık olarak kopyalanmasıdır, raporlama veri analistleri tarafından alınan raporlar incelenerek sistem stabilitesinden hangi amaçla kullanıldığına göre değişen raporlar alınabilmektedir. Bu farklı özelliklerin kullanımı için farklı uzmanlık alanları gelişmiştir.

Bir programcı SQL serveri verileri daha kolay yönetmek için kullanırken, bir veri analistcisi programcının yardımıyla gelen verileri raporlar haline getirerek çıkartımlarda bulunmak için kullanmaktadır. SQL server ile bir uygulama geliştiremez ancak farklı bir ortamda geliştirilen bir uygulama için veri deposu olarak kullanılır.

SQL server ile birlikte çok uyumlu çalışan Visual Studio gibi Microsoftun desteklediği bir dille kullanmak daha avantajlı hale gelmektedir. Eğer Microsoft SQL Server kullanıyorsa T-SQL dili kullanmak gerekmektedir. Şekil 2.7'de MSSQL veri tablosuna ilişkin bir örnek yer almaktadır.

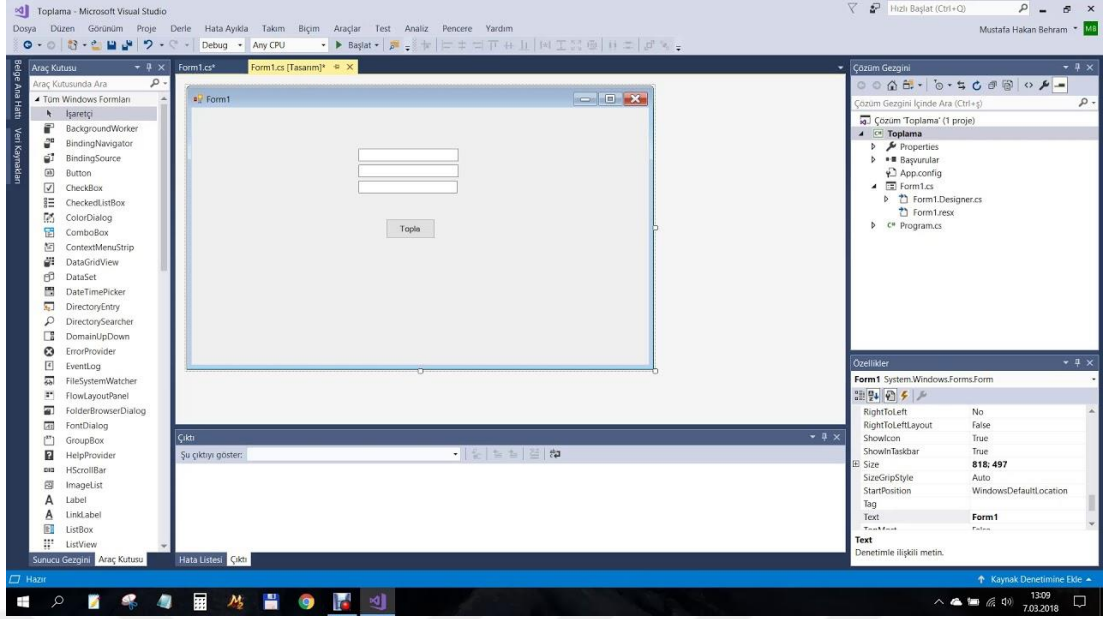
Last	FullName	CellPhone	MI	ID	First	MajorID	Deleted
Ramirez	Ramirez, Crisanto	(111) 22-3333	NULL	1	Crisanto	1	F
Koucouthakis	Koucouthakis, St...	(111) 22-3333	NULL	2	Steven	1	F
Warner	Warner, Michael	(111) 22-3333	NULL	3	Michael	1	F
Pate	Pate, Drew	(111) 22-3333	NULL	4	Drew	1	F
Pavlick	Pavlick, John	(111) 22-3333	NULL	5	John	1	F
Hicks	Hicks, Tom	(111) 22-3333	NULL	6	Tom	1	F
Francis	Francis, Robert H	(111) 22-3333	H	7	Robert	1	F
Agoston	Agoston, Laura H	(111) 22-3333	H	8	Laura	1	F
Akers	Akers, Lauren H	(111) 22-3333	H	9	Lauren	1	F
Peters	Peters, Joseph T	(111) 22-3333	T	10	Joseph	1	F
Aldana	Aldana, Joshua T	(111) 22-3333	T	11	Joshua	1	F
Smith	Smith, Joel T	(111) 22-3333	T	12	Joel	1	F
Alexander	Alexander, Reid T	(111) 22-3333	T	13	Reid	1	F
Alford	Alford, Kenneth T	(111) 22-3333	T	14	Kenneth	1	F
Bonnet	Bonnet, Jerrod T	(111) 22-3333	T	15	Jerrod	1	F
Macon	Macon, Britany T	(111) 22-3333	T	16	Britany	1	F

Şekil 2.7. MSSQL örnek veri tablosu

2.5. Visual Studio 2017

Microsoft tarafından üretilen ve desteklenen Visual Studio masaüstü uygulamalar, mobil uygulamalar, gömülü yazılımlar, web servis ve hizmetleri, sunucu hizmetleri gibi uygulamaları geliştirmek için kullanılabilir.

Visual Studio bir program yazma programıdır. Visual Studio'nun getirdiği basit ve kolay anlaşılabilen arayüzü, geliştiricinin işlerini kolaylaştırmak için tasarlanan araç çubuğu ve kütüphaneleri sayesinde hızlı bir şekilde ister web ister masaüstü uygulaması hızlı bir şekilde gerçekleştirilebilir. Şekil 2.8'de Visual Studio C# ile yazılmış basit bir uygulama ekranı gösterilmiştir.



Şekil 2.8. Visual Studio C# ile yazılmış bir uygulama

Visual Studio şık ve fonksiyonel bir arayüze sahip olduğu için programcının işini hızlandırarak projelere hız katar. Bazı fonksiyonları kısaca; hata ayıklama editörü, kodlama editörü, arayüz tasarlama editörü, veritabanı bağlantı ve ilişkilendirme editörü, geliştirilen uygulama için sürüm kontrolü gibi birçok özelliği bulunmaktadır.

Visual Studionun desteklediği başlıca programlama dilleri C, C++, C#, Visual Basic .NET, F#, Fossil, M, Python, HTML/XHTML/CSS, JavaScript'dir.

2.6. Python Derleyicisi

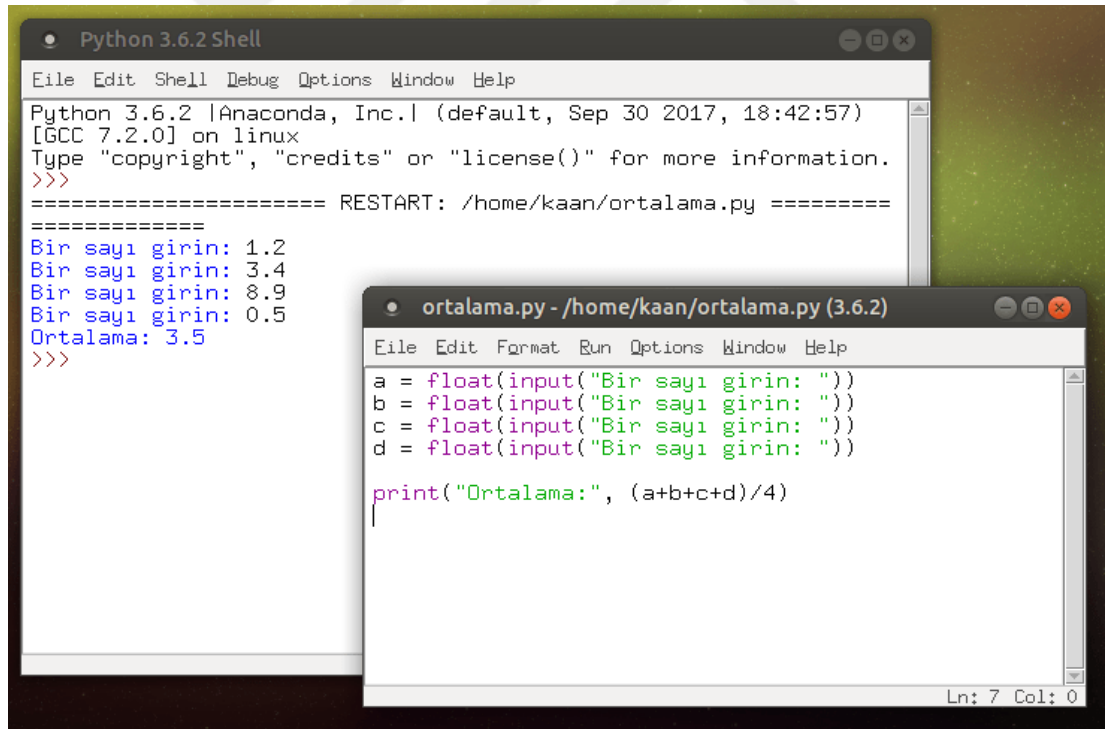
Python dili kullanım alanları; yapay zekâ, makine öğrenimi, sistem entegrasyonu, çeşitli otomasyonlar, web projeleri olarak sayılabilir.

Python dili geçmiş zamanlarda genellikle amacı olmayan uygulamalar için bir araç (exceldeki VBA macrolar gibi) olarak kullanılmaktaydı. Hatta popüler bir kitap olan sıkıcı şeyleri otomatikleştiren isimli bir kitapta bile Python dilinden bahsedilmiştir. Buna rağmen günümüzde Python dili modern bir yazılım olarak öne çıkmıştır. Python dili son zamanlarda bir yazılım geliştirme, veri analizi gibi faaliyetlerde büyük ölçüde tercih edilen bir dil konumuna gelmiştir. Artık bir araç olmaktan kurtulan Python dili üst düzey programcılarının dahi kullandığı bir dil olarak

kabul görmüştür. Uygulama geliştiriciler bu dili sistem yönetiminden web uygulamalarına kadar kullanmaktadır. Python programlama dili öyle bir hal almıştır ki bol örnek uygulama sayesinde yeni başlayanlar için bile kolay öğrenilebilen bir dil olmuştur.

Diğer programlama dillerini öğrenmek uzun zamanlar alır. Bunun sebebi karmaşık ve kullanım alanının büyük olmasıdır. Aynı anda birden çok alana (Web servisleri, web sayfaları, masaüstü uygulamaları, mobil uygulamalar, gömülü yazılımlar v.b.) hizmet ettikleri için öğrenmek hem uğraştırıcı hem zordur.

Python programlama kelime yapısıyla kolay anlaşılabilir. Kısa sürede hızlı bir şekilde uygulama geliştirilebilir ve devreye alınabilir. Günümüzde sürekli gelişmekte olduğundan hem kaynak bakımından hemde destek bakımından çok iyi durumdadır. Şekil 2.9’da Python dilinde yazılmış örnek bir kod ekranı verilmiştir.



```
Python 3.6.2 Shell
File Edit Shell Debug Options Window Help
Python 3.6.2 |Anaconda, Inc.| (default, Sep 30 2017, 18:42:57)
[GCC 7.2.0] on linux
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: /home/kaan/ortalama.py =====
>>>
Bir sayı girin: 1.2
Bir sayı girin: 3.4
Bir sayı girin: 8.9
Bir sayı girin: 0.5
Ortalama: 3.5
>>>

ortalama.py - /home/kaan/ortalama.py (3.6.2)
File Edit Format Run Options Window Help
a = float(input("Bir sayı girin: "))
b = float(input("Bir sayı girin: "))
c = float(input("Bir sayı girin: "))
d = float(input("Bir sayı girin: "))

print("Ortalama:", (a+b+c+d)/4)
Ln: 7 Col: 0
```

Şekil 2.9. Python dilinde yazılmış bir uygulama

Python dili ne kadar az duyulmuşsa da geliştiriciler arasında popüler bir kullanıma sahiptir. Github proje geliştirme ortamındaki projelerin büyük bir bölümü python tabanlıdır. Hatta Github, Index, Tiobe gibi proje geliştirme ve paylaşım

platformları arasında python tabanlı programlar günümüzde çok arandığı için en üst seviyelerde gözükmektedir.

Python mikroçiplerden sunuculara kadar çok geniş bir yelpaze'ye kullanılmaktadır ve desteklenmektedir. İnternetteki api sağlayıcı gibi servislerin çoğunda python tabanlı örnekler ve bilgiler mevcuttur. Bunun avantajı gelişmekte olan python diliyle uygulama geliştiren birisi için bir api desteğine ihtiyacı olduğunda örnek uygulamalardan yararlanarak hızlı ve güvenilir bir şekilde devreye alma gerçekleştirebilir.

2.6.1. Libbluet Kütüphanesi

Libbluet bluetooth kullanabilmek için geliştirilmiş bir java kütüphanesidir. İşletim sistemine özel bluetooth kütüphanelerini kullanarak Windows, Linux ve MacOX'te bluetooth protokolünü kullanımını sağlar. Libbluet Linux Modülü ve Libbluet Linux D-Bus Modülü olmak üzere iki farklı şekilde Linux'te bluetooth kullanımını sağlar. Bu modüllerin farklı harici kütüphanelere bağımlılıkları vardır.

2.6.2. Pip Kütüphanesi

PiP, Python'da third party library (üçüncü parti kütüphaneleri) kurulmasını sağlayan bir sistem. Python üzerinde geliştirme yaparken third party library'lere ihtiyaç duyduğunuzda hızlı bir şekilde kütüphaneleri eklemenize yardımcı bir kütüphanedir.

2.6.3. Pybluez Kütüphanesi

Pybluez kütüphanesi donanımınızda bulunan BLE modülü ile hızlı bir şekilde ibeacon taramaya izin verir ve bu paketleri yakalayarak size hızlı bir çözüm sunar.

3. MATERYAL VE YÖNTEMLER

3.1. Tanımlayıcı Etiket Tasarımı

HM-11 Modül iBeacon çalışma modunda çalıştırılarak yayın sıklığı 500ms olarak belirlenmiştir. Yayın sıklığının 500ms olması halinde pil olarak kullanılan CR2320 bataryasının 6 yıla kadar etiketi çalıştırabileceği hesaplanmıştır. Bu çalışma şartları altında tanımlayıcı etiketimiz çalışma süresi boyunca hiç durmadan UUID, MAJOR, MINOR, RSSI, TX POWER gibi parametreleri 500ms aralıklarla gönderebilecektir.

Tanımlayıcı etiketin çalışma süresi uzatılmak istenirse pil kapasitesi arttırabilir, paralel bir pil daha bağlanabilir veya yayın sıklığı süresi azaltılabilir. Tanımlayıcı etiketin kutusu Şekil 3.1. de gösterilmiştir. Etiket kutusu için bileğe bağlanması için kullanılan bileklik kordonu Şekil 3.2. de gösterilmiştir.



Şekil 3.1. Tanımlayıcı etiket kutusu



Şekil 3.2. Bileklik kordonu

3.1.1. HM-11 Modül Konfigürasyonu

HM-11 Modül parametreleri aşağıdaki gibi ayarlanmış olup bu iBeacon modunda 500ms yayın sıklığıyla çalışması içindir. TX Power:-53dBi, yayın sıklığı 500ms. HM-11 modülü beacon modunda çalıştırmak için mode parametresi 3 ayarlanmıştır. Yayın sıklığı 500ms ayarlandığında BLE modül 500ms de bir uyanıp

gerekli sinyalleri gönderdikten sonra derin uyku moduna geçerek güç tüketimini neredeyse sıfıra indirir.

3.1.2. Bağlantı Şeması

HM-11 Modül iBeacon modunda çalışır iken sadece besleme istemektedir. Ancak uygulama esnasında modül üzerinden parametre değişikliği yapabilmek için HM-11 BLE modülü derin uykudan uyandırmaya ihtiyaç vardır ve bunu reset pini ile yapabileceğimiz için reset pinine bir manyetik reed röle bağlanarak parametre değişikliği anında resetlemek için mıknatıs kullanılmıştır.

Modüle besleme girişi için 1 numaralı pinden pilin + ucu, 17 numaralı pinden ise pilin – ucu verilmiştir. Olası bir parametre değişiminde kutudan çıkarmakla uğraşmamak için reed rölenin bir ucu reset pinine diğer ucu – ye bağlanmıştır. Buda kutuyu açmadan bir mıknatısla modülü resetleyerek parametre değişikliğine izin vermektedir.

3.1.3. Taşıma Kılıfı

Taşıma kılıfı olarak kullanmak için HM-11 BLE Modül, CR2320 Pil ve Reed rölenin sığabileceği bir taşıma kutusu seçilmiştir. Ayrıca bileklik olarak kullanılmaya uygun olması açısından bileklik kordonu geçirmeye uygundur.



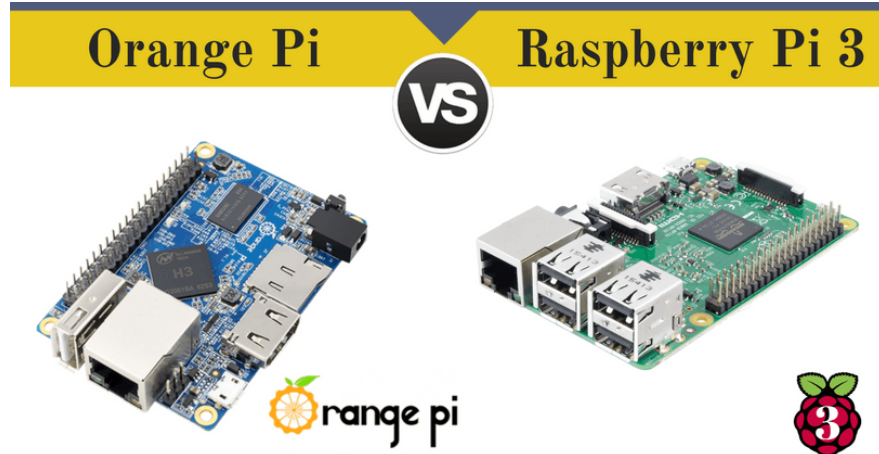
Şekil 3.3. Tanımlayıcı etiket tamamlanmış hali

3.2. Raspberry Pi 3 B+

Mini kart PC'ler arasından Raspberry Pi 3 B+ seçilmesindeki sebeplerden en önemlileri Raspberry Pi 3 B+'ın dahili BLE 4.0 modülü ile dahili kablosuz ağ adaptörü barındırması, programlama kolaylığı, birçok aksesuarın kolay temin edilebilirliği ve çeşitlerinin bolluğu, uygulama örneklerinin çok olması ve arıza anında hızlı devreye alınma kabiliyetidir.

Bu özelliklerine ekstra olarak personele bazı alanlarda veri girişi yaptırabilmek için dokunmatik bir LCD ekran bağlanabilmesi, görsel ve/veya sesli anons yaptırabilmek için dahili ses kartı bulunması, gerekli durumlarda harici aygıtlara sinyal gönderebilmek için giriş çıkış portları ve haberleşme protokolleri desteğininde büyük etkisi olmuştur.

Diğer kart bilgisayarlara kıyasla daha fazla kullanıcıya sahip olan Raspberry Pi modelleri donanımın yanı sıra yazılımsal özellikleriyle de öne çıkmaktadır. Kendine ait özel bir işletim sistemi bulduran Raspberry Pi, Rasbian işletim sistemi ile tam bütünleşmiş ve tüm özelliklerine erişilebilir halde kullanıma hazırlamak için bir SD karta ihtiyaç duymaktadır. Şekil 3.4'de Raspberry Pi'nin diğer kart bilgisayarı olan Orange Pi ile birlikte görseli verilmiştir.

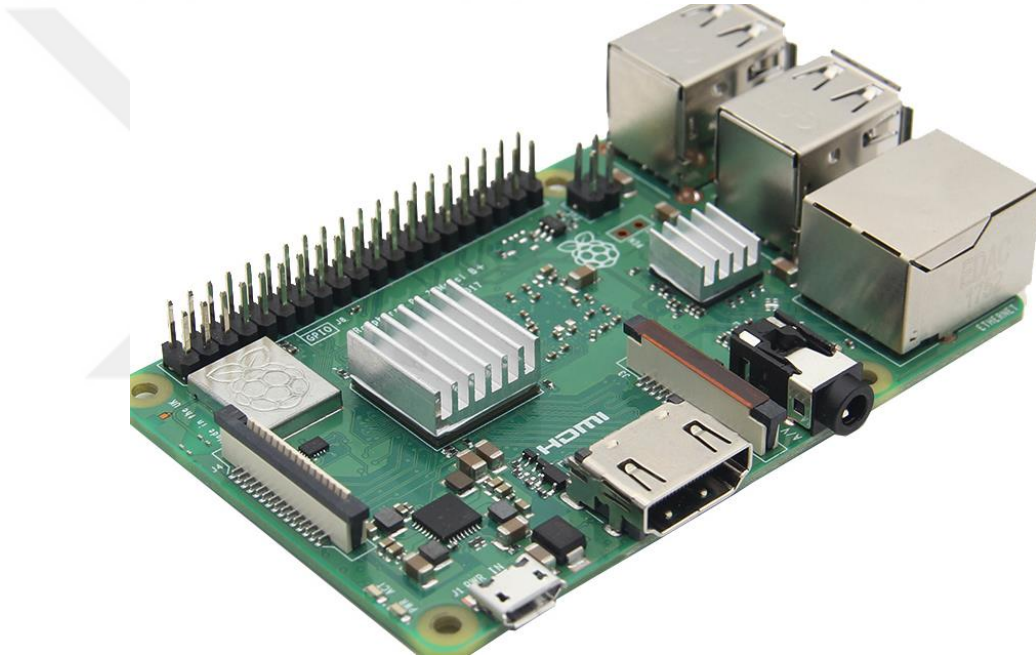


Şekil 3.4. Raspberry Pi 3 ve Orange Pi

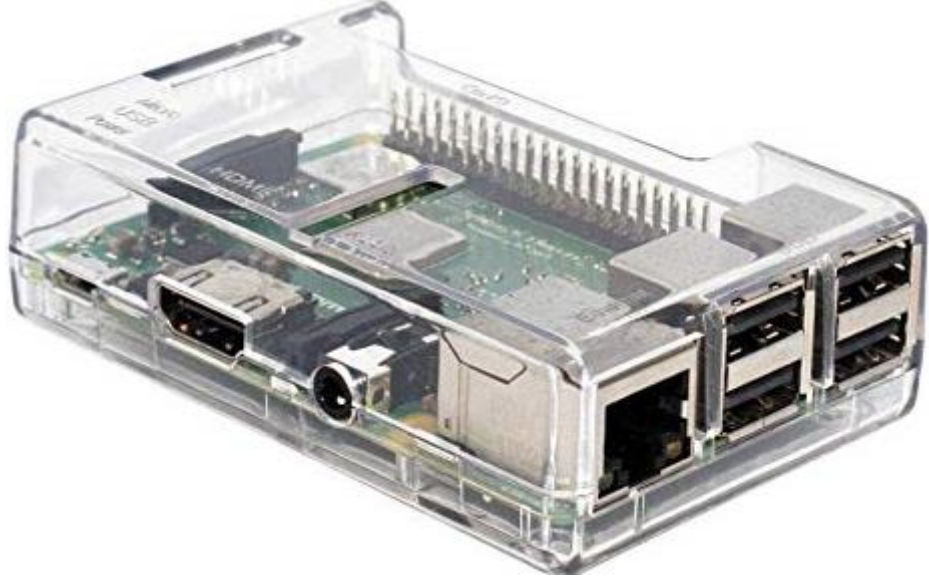
SD karta Raspbian görüntü dosyasını yazdırdıktan sonra Raspberry Pi çalıştırılabilir. Raspberry Pi’de uygulama geliştirmek için asgari olarak klavye, mouse, monitör üçlüsüne ihtiyaç vardır.

3.3 Algılayıcı Anten

Algılayıcı anten tasarımında Raspberry Pi 3 B+ kullanılmış olup Raspberry Pi 3 B+’ın dahili BLE 4.0 Bluetooth modülü kullanılmıştır. Raspberry Pi 3 B+ beslemesi güç için 5V 2,5A bir adaptör, özel olarak bu cihaz için üretilmiş bir şeffaf muhafaza kutusu ve işlemci, bağlantı modüllerinin soğutulması için aliminyum soğutucu kullanılmıştır. İşletim sistemini yüklemek için yüksek hızlı bir SD kart bağlanmıştır.



Şekil 3.5. Soğutuculu Raspberry Pi 3 B+



Şekil 3.6. Muhafaza kutusu ve Raspberry Pi 3 B+

3.3.1. Raspberry Pi 3 B+ Konfigürasyonu

İşletim sistemi olarak Rasbian, derleyici olarak Python derleyicisi kullanılmıştır. Python derleyicisinde kullanılan kütüphaneler; Libbluet, Pip, Pybluezdir. Raspberry Pi 3 B+ açıldığında beacon tarama uygulamasının çalıştırılması için işletim sistemi üzerine bir başlangıç scripti yazılmış olup bu script sadece gerekli uygulamaları sırasıyla başlatmaktadır.

3.3.2. Python

Python kullanım alanı çok geniştir, mühendislikten muhasebeye kadar birçok alanda kullanılır. 21. yüzyılın başlarında bilimsel hesaplamalar ve mühendislik çalışmalarında yoğun bir şekilde kullanılmaya başlanmıştır. Buna donanımsal ve yazılımsal güncellemeler sebep olmuştur.

Donanımsal olarak gelişen teknolojilerde işlem gücü, veri depolama ve haberleşme hızları arttığı için python gibi derlenmeden çalışan dillerin yorumlayıcıdan kaynaklanan yavaş olmaları ve kodların çok yer kaplaması gibi sorunlar göze batmayacak bir hal almıştır.

Yazılımsal olarak internet kullanımının yaygınlaşması ve dolayısıyla bilgiye kolay erişimin getirdiği avantajlarla açık kaynak kodlu özgür yazılım projelerine ilgi artmıştır.

Python kullanarak temel mühendislik ve bilimsel hesaplamalar için 3 kütüphane bulunmaktadır. Dizi işlemlerini hızlı gerçekleştirmek ve matris hesaplarında kullanmak için numpy, verilerin üzerinde sayısal işlemler yapmak, difarensiyel denklem, optimizasyon, entegrasyon, istatistik gibi ihtiyaçlar için scipy, grafik işlemleri görselleştirme gibi işlemler için matplotlib kütüphaneleridir. Bunların dışında ihtiyaç halinde kullanılması için C,R,F gibi dillerin python içinden çağırılabilmesi için aragüzler ve grafik işlemleri için kütüphanelerde mevcuttur. Şekil 3.7’de Python ile yazılmış mesafe hesaplama formül kodu yer almaktadır.

```
61  if mylist[2] == "4660" and mylist[3] == "64001":  
62     #print ("MAC: "+mylist[0])  
63     #print ("UUID: "+mylist[1])  
64     #print ("MINOR: "+mylist[2])  
65     #print ("MAJOR: "+mylist[3])  
66     #print ("TX POWER: "+mylist[4])  
67     #print ("RSSI: "+mylist[5])  
68     #accuracy = calculateAccuracy(mylist[4], mylist[5])  
69     distance = pow(10, ((float(mylist[4]) - float(mylist[5]))/20))  
70     #print ("DISTANCE:" + str(distance))  
71     #distance_filtered = distance#mov_avg(counterx, distance)  
72     #print ("DISTANCE: "+str(distance_filtered))
```

Şekil 3.7. Python ile mesafe hesaplama formülü

Python dili hızlı bir dil olmadığı için sistem çalışmasını etkileyecek hız isteyen projelerde kullanmak mantıksız olacaktır. Python ile hızın aşırı önemi olmadığı işlemlerin yapılarak sonuca ulaşıldığı ve görsel tasarımların ağır olmadığı uygulamalarda kullanmak daha doğru bir seçim olmaktadır.

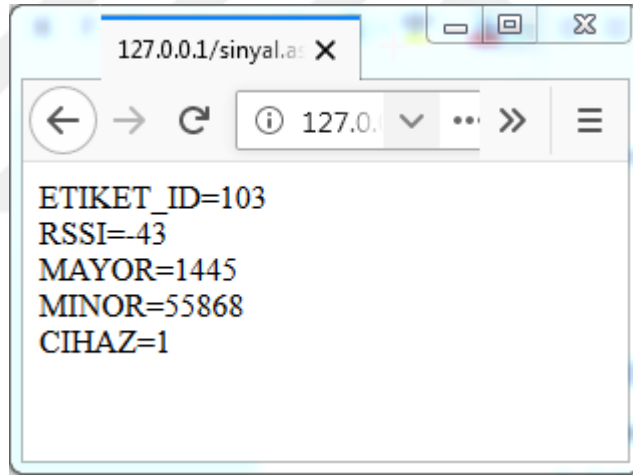
3.3.3. Yazılım Detayları

Python derleyicisinde ilgili kütüphaneler aracılığıyla MAC ID, UUID, MINOR, MAJOR, TX POWER, RSSI parametreleri tanımlayıcı etiketten alınmaktadır. Alınan veriler Python yazılımında mesafe bulma fonksiyonlarına girerek tahmini mesafe belirlenmektedir.

Yapılan çalışmada tahmini mesafenin doğruluğunu arttırmak için hareketli ortalamalar filtresi kullanılmıştır. Elde edilen filtrelenmiş mesafe bilgisi tanımlayıcı etiket numarası ile birlikte bütün parametreler web servisleri aracılığıyla sunucuya göndermek üzere hazırlanır. Sunucuya yazılan web servisi ASP Classic dilinde yazılmış olup gelen parametreleri sadece veri tabanına göndermekle yükümlüdür. Algılayıcı anten parametreleri;

http://192.168.1.4/sinyal.asp?ETIKET_ID=123&RSSI=-43&MAYOR=1445&MINOR=55868&CIHAZ=1

şeklinde bir URL ile sunucuya post etmektedir. Sunucu bu isteğe yanıt olarak parametreleri doğrulama amaçlı geri göndermektedir. İşlem sonucunda algılayıcı antenden alınan parametreler MSSQL Server'a yazılmıştır.

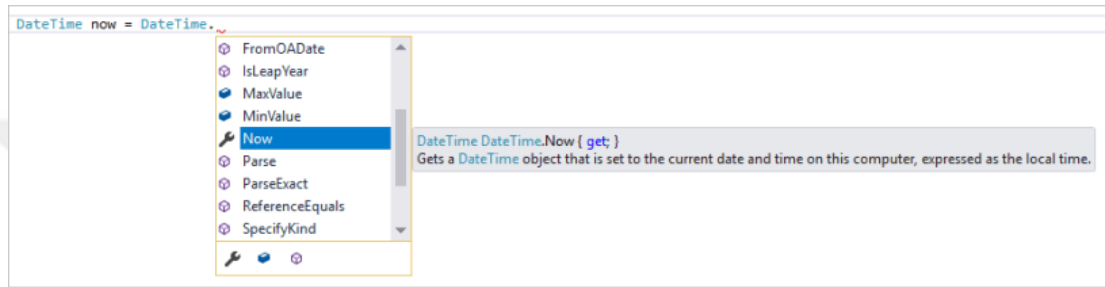


Şekil 3.8. Örnek post cevabı

3.4. Visual Studio 2017

Visual Studio Microsoft tarafından geliştirilen bir yazılım geliştirme ve yayınlama platformudur. Bu platformun 3 versiyonu bulunmaktadır: Community; öğrenciler, açık kaynak kodlu proje yapanlar ve bireysel kullanımlar için ücretsiz versiyonudur. Professional; ticari amaçlı küçük ekipler için en uygun seçenektir. Enterprise; yine ticari amaçlı büyük yazılım geliştirme firmaları için uygun seçenektir ve Professional'den farkı aynı projede aynı anda birçok kişinin çalışabilmesine olanak vermesidir.

Visual studio ile uygulama geliştirebilir, geliştirilen uygulama düzenlenebilir, derlenebilir ve çok kolay bir şekilde paketlenir. İçerisinde barındırdığı özellikler sayesinde birçok uygulamayı basit, hızlı ve sorunsuz geliştirebiliriz. Bu özelliklerden bazıları; RE Factoring kod düzenleme ve değişkenlerin yeniden adlandırılması esnasında birden fazla satırın aynı anda değiştirilmesi için gerekli işlemleri yapabilen bir özelliktir, IntelliSense kod yazarken programcıya küçük ipuçları, hatırlatmalar ve uyarılar veren bazı durumlarda da küçük kod parçaları için programcıya yardımcı olan bir özelliktir.

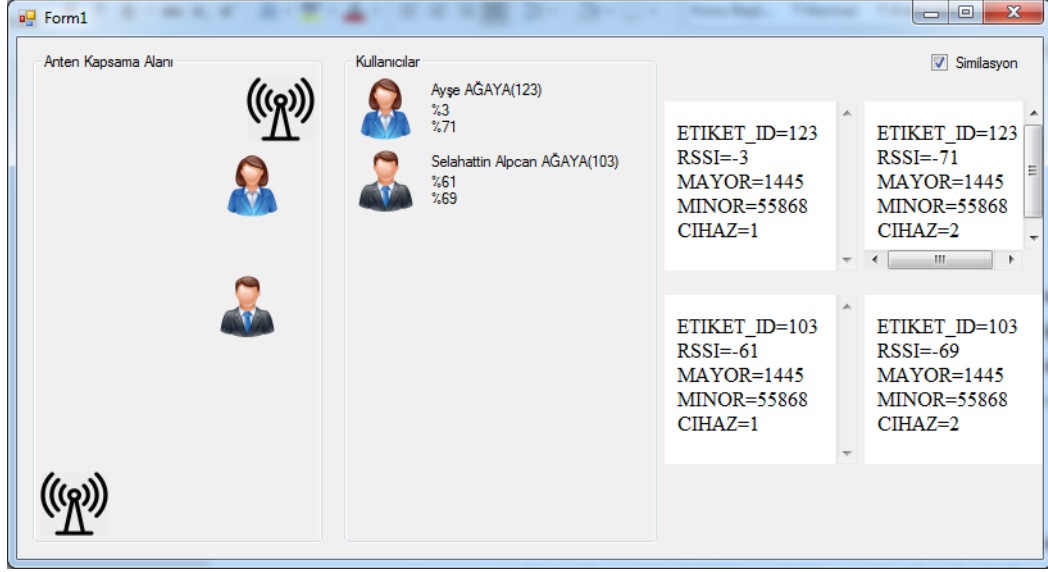


Şekil 3.9. IntelliSense için örnek

Quick Launch uygulama yazarken gerekli olan herşey bu araç yardımıyla bulunabilir. Squiggles and Quick Actions program kodundaki hataların ve uyarıların altını durumuna göre çeşitli renklerde dalgali çizgi ile çizer ve hatanın yerini bulmada programcıya kolaylık sağlar.

3.5. Arayüz Tasarımı

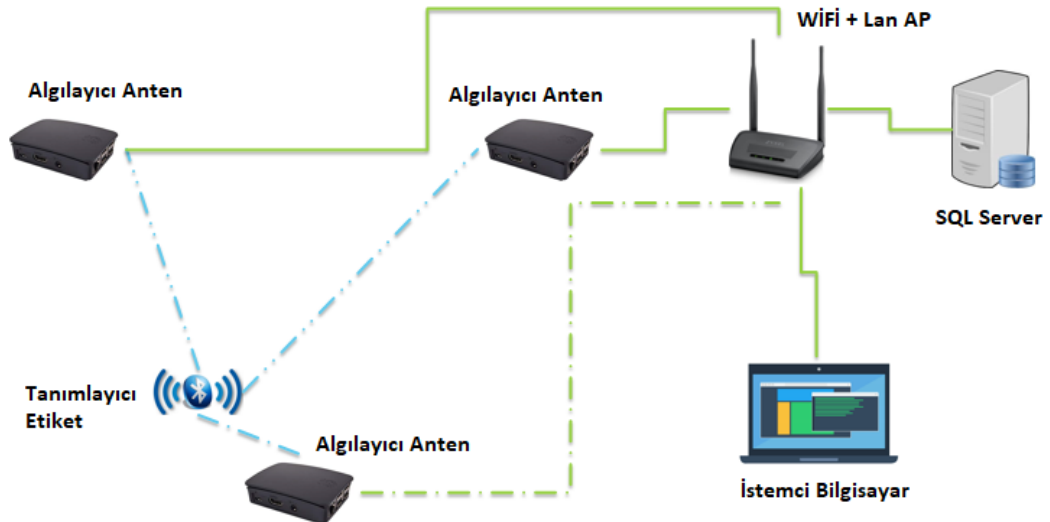
Projede kullanılan arayüz programı Visual Studio 2017 derleyicisinde C# dilinde yazılmıştır. Arayüz programı MSSQL Server'a bağlanarak konum bilgilerini zaman damgasıyla birlikte analiz ederek anlık olarak haritada göstermektedir. Tanımlayıcı etiket 1 dakikadan fazla süredir herhangi bir Algılayıcı Antene sinyal iletmezse kapsama alanı dışında olduğu için haritadan kaybolmaktadır. Şekil 3.10'da iki personelin tasarlanan arayüzde konumları gösterilmiştir.



Şekil 3.10. Arayüz ekran görüntüsü

3.6. Sistem Topolojisi

Sistemde kullanılan tüm algılayıcı antenler ve sunucu aynı ağda olmalıdır. Bu ağ yapısının oluşturulması için kablolu ve kablosuz ağ atlatıcılar kullanılmıştır. Algılayıcı antenler ağa hem kablosuz hemde kablolu olarak katılabilmektedir. Algılayıcı antenin biri kablolu olarak bağlanırken diğeri kablosuz olarak ağa katılabilmektedir. Ayrıca tanımlayıcı etiketler ve algılayıcı antenler arasında tek yönlü bir bluetooth ağı oluşmaktadır bu ağ sadece parametrelerin iletilmesi amacıyla kullanılmaktadır.



Şekil 3.11. Sistem topolojisi

3.7. Microsoft SQL Server Express

SQL kısaltması, Structured Query Language cümlesinden gelir ve Türkçe karşılığı Yapısal Sorgulama Dili'dir. Veritabanları için kullanılan standart bir dildir. Sadece bazı veri tabanı yapılarında küçük değişiklikler görülebilir. Bu değişikliklere de şive farkı olarak bakmak daha doğru olacaktır, yapısal bütünlüğü bozmayan küçük değişikliklerdir.

Bazı veritabanı sistemleri, Oracle, MySQL, MsSQL, Informix, Postgres, Firebase'dir. Ancak veri yönetimi için bir veritabanı kullanacağımızda özel bir isteğimiz yoksa 2 popüler veri tabanı sistemi öne çıkar. Bunlar; MySQL ve MsSQL'dir.

MySQL veritabanı sistemi GNU genel kamu lisansı ile lisanslanmış olup, açık kaynak kodlu ücretsiz bir yapıdır. MsSQL Server ise Microsoft tarafında üretilmiştir. Microsoft SQL Server olarak bilinir. Ancak, MySQL gibi ücretsiz değildir. Küçük projeler için MsSQL Express isimli bir ücretsiz versiyonu bulunmaktadır. Ücretsiz versiyonun birçok özelliği de kısıtlıdır.

MySQL'in en iyi performans gösterdiği ortamlar genellikle veri depolama ve referans verileri için hazırlanmış ortamlardır. Ancak veri ekleme ve silme işlemlerinde bu performansı gösteremez. Dolayısıyla veri ekleme ve silme işlemlerinde MsSQL bu konudaki farkını ortaya koyarak bir adım öne geçmiştir.

Veritabanı olarak kullanılan Microsoft SQL Server veri yönetimindeki kolaylığı ve arkasında kurumsal desteği olan bir veritabanı olduğu için bu çalışmada Microsoft SQL Server kullanılması kararlaştırılmıştır.



Şekil 3.12. MySQL ve MsSQL

3.8. Veritabanı Tasarımı

Projede veri tabanı olarak Microsoft firması tarafından geliştirilen MsSQL Server kullanılmıştır. MsSQL Server üzerinde veriler iki tabloda tutulmakta olup bu tablolarda aşağıdaki bilgiler saklanmaktadır.

Tablo 3.1. Sinyaller tablosu örnek veriler

ID	ETIKET_ID	RSSI	MAYOR	MINOR	CIHAZ	TARİH
2	123	-43	1445	55868	1	29.07.2019 11:26
3	123	-63	1445	55868	1	29.07.2019 11:47
4	123	-63	1445	55868	1	29.07.2019 11:57
5	123	-83	1445	55868	1	29.07.2019 11:57
6	103	-83	1445	55868	1	29.07.2019 11:57
7	103	-73	1445	55868	1	29.07.2019 11:57
8	103	-43	1445	55868	1	29.07.2019 11:57
9	103	-43	1445	55868	1	29.07.2019 12:19
10	123	-43	1445	55868	1	29.07.2019 12:19
11	123	-43	1445	55868	1	29.07.2019 12:23

Tablo 3.2. Personeller tablosu örnek veriler.

ID	ADI_SOYADI	CEP	ETIKET_ID
1	A	5555555555	103
2	B	5555555555	123

Veritabanında canlı izleme için bir görünüm oluşturulmuş olup tanımlayıcı etiketlerden gelen sinyalleri geçmişe dönük 1 dakikalık sürelerle okuyarak canlı haritalama amacıyla yazılmıştır. Bu görünüm arayüz programının canlı haritalandırma bölümünde kullanılmaktadır.

Verilerin analizi arayüz programında da yapabildi. Ancak bu işlem yükünü istemci bir bilgisayardan ziyade güçlü bir Server'a yaptırmak daha avantajlı olacağı için bu yöntem tercih edilmiştir. Görünüm'in oluşturduğu veri seti aşağıdaki gibidir.

Tablo 3.3. Canlı izle görünüm örnek veriler.

ID	ETIKET_ID	RSSI	MAYOR	MINOR	CIHAZ	TARİH
631	123	-61	1445	55868	1	31.07.2019 14:34
633	103	-83	1445	55868	1	31.07.2019 14:34
632	123	-81	1445	55868	2	31.07.2019 14:34
634	103	-64	1445	55868	2	31.07.2019 14:34

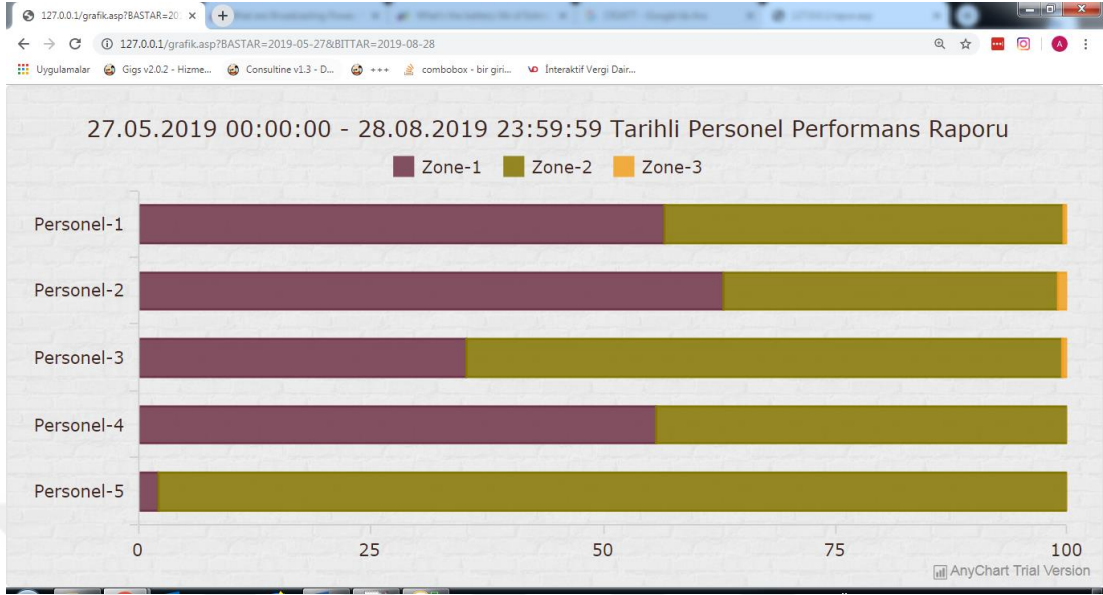
Personel bilgilerinin tutulduğu tabloda; personel numarası *int* tipi değişkende saklanmaktadır ve otomatik olarak benzersiz verilmektedir, adı ve soyadı *nvarchar(MAX)* tipi değişkende tutulmaktadır ve arayüz programından girilmektedir, cep telefonu *nvarchar(MAX)* tipi değişkende saklanmakta olup yine arayüz programından girilmektedir, son olarak eşleştirme için kullanılacak etiket numarası *nvarchar(MAX)* tipi bir değişkende tutularak arayüz programından girilmektedir.

Sinyallerin tutulduğu tabloda; sıra numarası *int* tipi değişkende tutularak otomatik olarak benzersiz bir sayı almaktadır, etiket numarası *nvarchar(MAX)* tipi değişkende tutularak tanımlayıcı anten tarafından web servisleri aracılığıyla iletilmektedir, RSSI sinyal kalitesi *int* tipi değişkende tutularak algılayıcı anten tarafından web servisleri aracılığıyla iletilmektedir, mayor *int* tipi değişkende tutularak algılayıcı anten tarafından web servisleri aracılığıyla iletilmektedir, minor *int* tipi değişkende tutularak algılayıcı anten tarafından web servisleri aracılığıyla iletilmektedir, cihaz numarası *int* tipi değişkende tutularak algılayıcı anten tarafından web servisleri aracılığıyla iletilmektedir, son olarak tarih damgası *datetime* tipi değişkende tutularak web servisinin çalıştığı andaki sistem tarih saati değerini almaktadır.

3.9. Web Tabanlı Raporlama Arayüzü

Projede web tabanlı bir performans raporu hazırlanmıştır. Rapor hazırlanırken HTML, CSS, JAVASCRIPT, ASP Classic dilleri kullanılarak hazırlanmış olup veritabanı tarafında verileri rapor için uygun hale getiren bir görünüm kullanılmıştır. Rapor için her bir veya birden fazla algılayıcı anten için bir alan belirlenmelidir. Rapor personelin bu alanlara giriş ve çıkış yaptığı saatlere göre hesaplanarak, ilgili personelin gün, hafta veya ay içersinde kaç saat üretim alanında veya kaç saat dinlenme alanında

geçirdiği gibi bilgileri grafik haline getirerek göstermektedir. Arayüz tasarımı aşağıda gösterilmiştir.



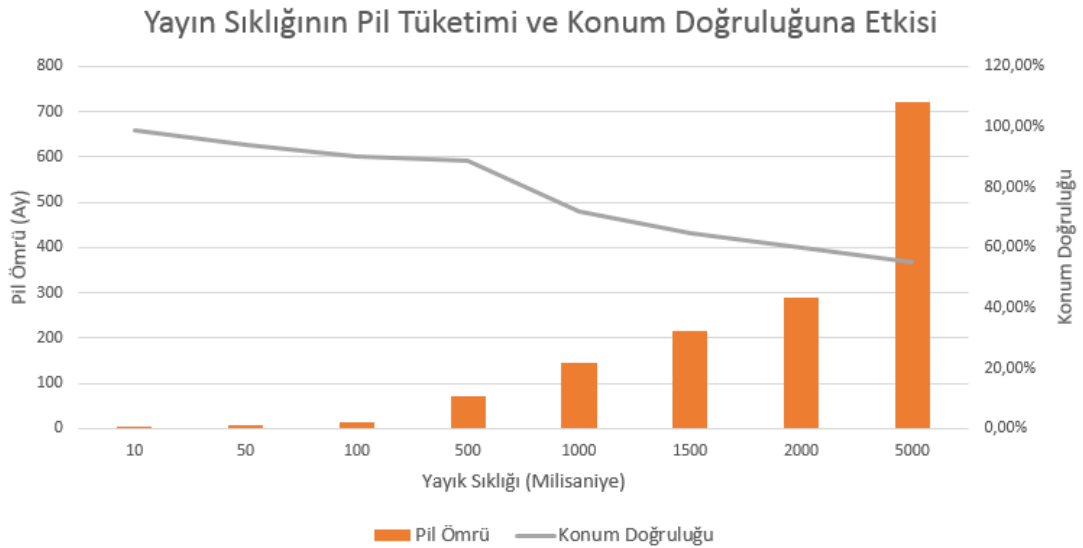
Şekil 3.13. Web tabanlı raporlama arayüzü tasarımı masaüstü görünümü

4. ARAŞTIRMA BULGULARI VE TARTIŞMA

Bu bölümde çalışmadan elde edilen verilerin analizi, karşılaşılan sorunlar ve çözümleri yer almaktadır. Öncelikle süreklilik testleri uygulanarak kararlılık sağlanmıştır. Sonrasında güç tüketimi, mekanik tasarımlar ve yazılım mantığı oluşturulmuştur.

4.1. HM-11 BLE Modül Testleri

HM-11 modülün testleri anında yayın sıklığının belirlenmesinde her biri aynı konumdan olmak üzere alınan 250 sinyal ile yapılmıştır. Yapılan testler sonucunda Şekil 4.1'deki değerlere göre en uygun değer 500ms yayın aralığı olduğu belirlenmiştir. Bu değer belirlenirken modülü kullanacak kişilerin pil değiştirme süreleri ve konum doğruluğu dikkate alınmıştır. Şekil 4.1'de HM-11 modülünde kullanılan entegrenin güç tüketimine göre bir hesaplama grafiği oluşturulmuştur.



Şekil 4.1. Yayın sıklığının pil tüketimi ve konum doğruluğuna etkisi

Yayın sıklığı belirlendikten sonra gerçek bir ortamda elektriksel gürültü, insan faktörü gibi parametreleri de test edebilmek için dış ortam testine tabi tutulmuştur. Testler sonucunda ortamdaki insan sayısı ve elektriksel gürültü arttıkça konum tespitinde sapmalar meydana gelmekteydi. Bu sorunu çözebilmek için hareketli ortalamalar filtresi denemeleri yapılmıştır.

Hareketli ortalamalar filtresinde belirli bir süre boyunca veri toplanır ve gelen veriler programcının istediği boyuta ulaşınca ortalama hesabı yapılır ve sonrasında sonuç değeri aktarılır. Hareketli ortalamalar filtresi kullanılırken ortalaması alınacak veri ne kadar çok ise hata toleransı o kadar düşük olacaktır.

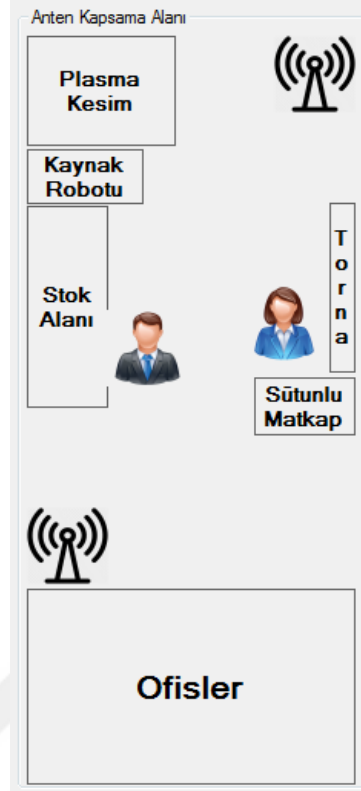
Ayrıca hareketli ortalamalar filtresinin normal ortalamaya göre avantajı toplanacak veri sayısı 10 iken normal ortalamada her 10 veri geldiğinde bir ortalama alınırken hareketli ortalamalar filtresinde ilk 10 veri geldikten sonra gelen her veri için ortalama değeri çıkartmasıdır. Burada ilk 10 veriden sonra 11. veri geldiğinde 1. sıradaki veriyi hesaplama dışına çıkartarak yapmaktadır.

Hareketli ortalamalar filtresinin kullanıldıktan sonra aralıklarla kaybolan konum kaybolmamaya ve daha düzgün bir konumlandırma yapılmasına olanak sağladığı görülmüştür. Bu durum personel takibini daha kararlı bir hale getirmiştir.

4.2. Yazılım Tasarımı ve Testleri

Yazılım tasarımı yapılırken Visual Studio 2017 ortamında C# dili kullanılmıştır. Ayrıca kullanılan kütüphaneler; System, System.Collections.Generic, System.ComponentModel, System.Data, System.Data.SqlClient, System.Drawing, System.Linq, System.Text, System.Threading.Tasks ve System.Windows.Forms.

Arayüz tasarımı yapılırken iki algılayıcı antenli bir alan düşünülmüştür. Bu alanda veritabanından gelen sinyal kalitesine göre basit bir haritalandırma yapılmaktadır. Şekil 4.2’de personellerin arayüz canlı izleme alanındaki konumları görülmektedir.



Şekil 4.2. Arayüz canlı izleme alanı

Arayüz tasarımı belirlendikten sonra antenler arası mesafelerin gerçek ortamdaki karşılığına oranını bulabilmek için öncelikle antenler arası piksel sayısı ölçülmüş ve ardından gerçek ortamdaki elde edilen verilerle oranlanarak görsel tasarım arayüzü tasarlanmıştır.

Yapılan oranlamaları test etmek amacıyla bir benzetim ortamı hazırlanmıştır. Bu benzetim ortamı her bir personelin her iki algılayıcıyada rastgele değer göndermesi için oluşturulmuştur. Veriler Visual Studio'da webbrowser elementi aracılığıyla gönderileceği için 4 adet web browser yerleştirilmiş ve simülasyon timer'ını başlatan bir checkbox'a bağlanmıştır. Simülasyon timer'ı başlatıldığında her 5sn'de bir random üretilen konum bilgileri http://127.0.0.1/sinyal.asp?ETIKET_ID=123&RSSI=-55&MAYOR=1445&MINOR=55868&CIHAZ=1 benzeri bir URL'ye dönüştürülerek webbrowserler aracılığıyla post edilmiş ve MsSQL server'a kaydı yapılmıştır.

MsSQL Server'dan verileri okuyan Visual Studio ortamında hazırlanan arayüz programımız verilerin tamamını okuyarak zaman damgası ile birlikte analiz ettikten sonra canlı izleme için hazırlayıp gerekli fonksiyonları başlatmaktaydı. Ancak bir süre

sonra veriler çoğaldıkça veri tabanından tüm verilerin alınması zaman alır bir hale geldi ve canlı izlemede gecikmeler, donmalar yaşanmaya başladı. Donmaların kaynağını araştırırken MsSQL serverla aradaki trafiğin çok büyüdüğünü ve gittikçe dahada büyüyeceği görüldü ve canlı izleme yaparken okunan tüm verilerin arayüz programında değil bu iş için oluşturulan MsSQL serverda yapmanın daha uygun olacağı kararlaştırıldı.

Sonrasında MsSQL server'a bir görünüm yazmaya karar verilerek çalışmalara başlandı. Yazılan görünüm 1 dakikadaki en güncel konum verisini yollaması için öncelikle son 1 dakika harici diğer veriler filtrelenerek seçimden çıkartıldı ve sonrasında zaman damgasına bakılarak son bir dakikada gelen birden fazla veri var ise bunların en güncel olanını seçerek arayüz programı için hazırlandı. Bu işlemi her bir personel için yaparak canlı izleme ekranında personellerin konumlarının gösterilmesi sağlandı.

Yapılan işlem sonrasında arayüz programındaki yavaşlama ve donmalar ortadan kalkmış olup canlı izleme esnasında herhangi bir olumsuzlukla karşılaşılmamıştır.

4.3. Tanımlayıcı Etiket Tasarımı

Tanımlayıcı etiket ilk olarak kart tipi tasarlanacaktı ancak hem maliyet açısından hemde zaman açısından projeye uygun olmadığı için daha basit bir çözüm olan bileklik olarak tasarlanmıştır.

Bileklik olarak tasarlama aşamasında öncelikle pil seçimi yapmak gerekiyordu zaten düşük güç tüketimli bir modül kullandığımız için burada kapasiteden ziyade boyutlardan kazanmak için 3.3V CR2320 pil kullanılmıştır.

Pil seçiminden sonra kutu seçiminde öncelikle pil ölçüleri HM-11 ölçüleri gibi etkenleri göz önünde bulundurarak bileklik yapımına uygun çift taraflı kulakları olan bir kutu seçimi yapıldı. Bilekliğin kolayca taşınabilmesi için bir saat kordonu yapısı kullanılmıştır. Son olarak bu tasarım yapısı sonrasında sahadaki sinyalde zayıflama olup olmadığı kontrol edildikten sonra bileklikler kullanıma hazır hale getirildi.

4.4. Saha Testleri

WRS Robotik İth. İhr. San. Ve Tic. Ltd. Şti. şirketinde gerçekleştirilen testler doğrultusunda personelin devam takibi sistem aracılığıyla yapılmaktadır. Ancak bazı durumlarda örneğin işletmede değilde sahada çalışan personeller için manuel giriş yapılması gerekmektedir. WRS Robotik firmasında Ofislerin bulunduğu alana bir adet ve atölyenin bulunduğu alana bir adet algılayıcı anten konumlandırılarak uygulama yapılmıştır. Antenler kurulurken özellikle ilgili belirlenen alan içerisinde her noktadan kolay sinyal alabilecek şekilde yerleştirilmiştir.



Şekil 4.3. Ofis algılayıcı anten konumu



Şekil 4.4. Atölye algılayıcı anten konumu

Yapılan testler sonucunda 2 algılayıcı anten ile personelin kesin konum tespiti yapılamamıştır. Sadece belirli alanlara giren ve çıkan personel kayıtları tutulabilmektedir.

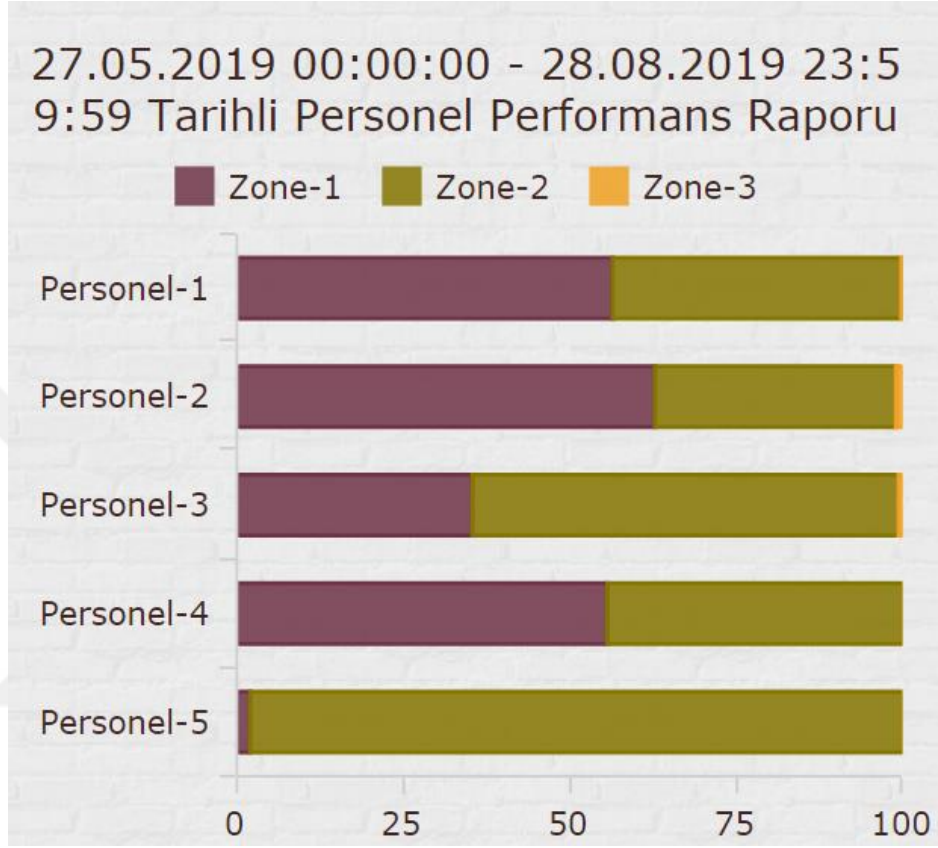
GPS gibi konumlama teknolojilerinde konum tespiti için en az 3 sinyal kaynağından sinyal alınması gerekmektedir. 2 kaynaktan alınan sinyal ile yalnızca

bölge hesaplanabilmektedir. Çalışmayı ilerleterek 2 anten ile sadece düz bir koridorda tahmini konum tespiti yapılmıştır. Arayüz programı da buna göre düzenlenmiştir. Tablo 4.1’de saha uygulamasından elde edilen bir günlük veriler yer almaktadır. Veri toplama işleminde işletmede bulunan 5 personel için PKDS sistemi uygulanmıştır. Buna göre personelin işletmenin hangi alanında ne kadar süre ile buldukları haritalaması raporlanabilmektedir. Yapılan testler sonucunda işletme sınırları içerisine giren bir personelin sunucu tarafından algılanması 10sn civarında bir gecikme ile gerçekleşmektedir.

Tablo 4.1. Personel Devam Kontrol Tablosu Bir Günlük Örnek

ETIKET_ID	CIHAZ_ID	GIRIS_ZAMAN	CIKIS_ZAMAN	SURE
505	1	2019-07-18 07:25:47	2019-07-18 07:35:36	00:09:49
505	2	2019-07-18 07:37:21	2019-07-18 18:35:59	10:58:38
502	1	2019-07-18 08:31:35	2019-07-18 12:01:45	03:30:10
503	2	2019-07-18 08:31:45	2019-07-18 18:32:55	10:01:10
504	2	2019-07-18 09:01:29	2019-07-18 13:55:12	04:53:43
501	1	2019-07-18 09:11:25	2019-07-18 12:11:45	03:00:20
502	1	2019-07-18 12:31:45	2019-07-18 18:31:38	05:59:53
504	1	2019-07-18 13:59:42	2019-07-18 14:47:34	00:47:52
501	2	2019-07-18 18:25:28	2019-07-18 18:35:51	00:10:23
504	2	2019-07-18 14:49:21	2019-07-18 18:39:26	03:50:05
505	1	2019-07-18 07:14:12	2019-07-18 07:20:26	00:06:14
501	1	2019-07-19 07:22:14	2019-07-19 15:45:55	08:23:41
501	2	2019-07-19 16:21:15	2019-07-19 18:45:49	02:24:34
502	1	2019-07-19 07:27:45	2019-07-19 11:46:52	04:19:07
502	2	2019-07-19 12:19:34	2019-07-19 18:45:21	06:25:47
503	1	2019-07-19 07:39:35	2019-07-19 18:35:33	10:55:58
504	1	2019-07-19 07:21:17	2019-07-19 11:45:24	04:24:07
505	1	2019-07-20 07:23:24	2019-07-20 07:37:46	00:14:22
505	2	2019-07-20 07:39:36	2019-07-20 18:32:12	10:52:36
502	1	2019-07-20 08:31:35	2019-07-20 12:01:45	03:30:10
503	2	2019-07-20 08:28:44	2019-07-20 18:22:15	09:53:31
504	2	2019-07-20 08:41:12	2019-07-20 15:55:12	07:14:00
501	1	2019-07-22 08:31:25	2019-07-22 12:11:45	03:40:20
502	1	2019-07-22 08:21:45	2019-07-22 18:31:38	10:09:53
504	1	2019-07-22 08:24:12	2019-07-22 13:32:11	05:07:59
501	2	2019-07-22 13:25:28	2019-07-22 18:35:51	05:10:23
504	2	2019-07-22 14:47:34	2019-07-22 18:39:26	03:51:52
501	1	2019-07-23 09:11:25	2019-07-23 12:11:45	03:00:20
502	1	2019-07-23 12:31:45	2019-07-23 18:31:38	05:59:53
504	1	2019-07-23 13:59:42	2019-07-23 11:47:34	21:47:52
501	2	2019-07-23 15:25:28	2019-07-23 18:28:49	03:03:21

Şekil 4.5’de Tablo 4.1’de yer alan veri tabanı verilerine göre geliştirilen yazılım tarafından oluşturulan mobil arayüz yer almaktadır. Buna göre, hangi personelin tesis içerisindeki hangi alanda ne kadar süre geçirdikleri bar grafik şeklinde görselleştirilmiştir.



Şekil 4.5. Web tabanlı raporlama arayüzü tasarımı mobil görünümü

5. SONUÇ VE ÖNERİLER

Proje sonucunda BLE 4.0 teknolojisi kullanılarak fabrika içerisinde personel konumlandırma işlemi test ortamında başarıyla gerçekleştirilmiştir. Tanımlayıcı etiketin kutusu bileklik şeklinde tasarlanarak basit, ucuz ve ergonomik bir bileklik elde edilmiştir.

Algılayıcı anten Raspberry Pi 3 B+ kullanılarak dahili BLE ve kablosuz ağ adaptörleri aracılığıyla toplanan tanımlayıcı etiket verileri anlık olarak veritabanına gönderilmektedir. Gönderilen verileri veritabanında yazılan görünüm canlı izleme için son hale getirerek arayüz programı aracılığıyla son kullanıcıya personelin anlık konumunu göstermektedir.

Çalışmanın görsel arayüzlerinde personelin tesis içerisindeki hareket alanı çıkarılmış, hangi bölümlerde ne kadar zaman geçirdiğiyle ilgili veriler toplanmıştır. Bu verileri tam anlamıyla incelemek büyük bir sunucu ve test ortamı gerektirdiği için detaylı raporlamalar oluşturulmamıştır. Proje tam anlamıyla tesis ortamında test edilmediği için performansa katkısı gözlemlenememiştir.

Tesis içindeki personellerin konumları bilindiği için yetkilendirme ve raporlama üzerine araştırma yapılabilir. Fabrikalardaki üretim takip yazılımı olan MES yazılımları ile bu çalışmada kullanılan yöntem ve elde edilen bulgular göz önüne alınarak personellerin üretim performansları ve personelin kullandığı ekipmanın verimliliği hesaplanabilir. Bu çalışmanın devamında, BLE 5.0 kullanılması ile konum bulma daha hassas olarak gerçekleştirilebilir.

KAYNAKLAR

- [1] Abdulkareem, Ahmed. Bluetooth düşük enerji beaconlarını kullanarak akıllı bir yoklama takip sisteminin tasarımı, geliştirilmesi ve değerlendirilmesi. Çankaya Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı, Ankara, 2018, 1. (Yüksek Lisans Tezi).
- [2] Aydalka, Sevgi Cansu. Bluetooth beacon teknolojisi kullanarak lokasyon ve müze eser tanıtım uygulaması gerçekleştirilmesi. Balıkesir Üniversitesi, Fen Bilimleri Enstitüsü, Elektrik-Elektronik Mühendisliği Anabilim Dalı, Balıkesir, 2018, 1. (Yüksek Lisans Tezi).
- [3] Masmanacı, Mehmet Nazif. Bluetooth beacon temelli kapalı alan mobil konum tespit ve analiz sistemi geliştirilmesi. Erciyes Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı, Kayseri, 2018, 1. (Yüksek Lisans Tezi).
- [4] Bayılmış, C., Özdemir, M. Bluetooth Düşük Enerji Teknolojisine Sahip İşaretçi ve Akıllı Telefon Temelli Öğrenci Yoklama Sistemi. Bilişim Teknolojileri Dergisi. 2016, Cilt(9-3), 249–254.
- [5] İlkuçar, M. Sensör Tabanlı Yerleşke Otomasyon Sistemi: İbeacon Teknolojileri Yardımı İle Personel Bilgi Mobil Uygulaması. International Science And Technology Conference, July 13-15, 2016 Vienna-Austria.
- [6] Küçük, K., Çelik, B., Bayılmış, C. Nesnelerin İnterneti Teknolojileri ile Gerçek Zamanlı Okul Servisi ve Öğrenci Takip Sistemi Tasarımı. Düzce Üniversitesi Bilim ve Teknoloji Dergisi Cilt(6-2018),
- [7] Lin, X., Ho, T., Fang, C., Yen, Z., Yang, B., Lai, F., A mobile indoor positioning system based on iBeacon technology, 2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), 2015, DOI: 10.1109/EMBC.2015.7319507
- [8] Yang, J., Wang, Z., Zhang, X. An iBeacon-based Indoor Positioning Systems for Hospitals., International Journal of Smart Home, Cilt 9-7,161-168
- [9] Arsan, T., Büyük Patlama – Büyük Çöküş Optimizasyon Yöntemi Kullanılarak Bluetooth Tabanlı İç Mekan Konum Belirleme Sisteminin Doğruluğunun İyileştirilmesi, Süleyman Demirel Üniversitesi Fen Bilimleri Enstitüsü Dergisi, 2018, Cilt 22, 367-374
- [10] Aldemir, O., Dinçer, Z., Aydın, E., Öktem, R., Aktif RFID Sistemi Kullanarak İç Mekanda Yönlendirme, URSI – Türkiye 2006 3. Bilimsel Kongresi, 6-8 Eylül 2006.
- [11] Ayabakan, T., Kerestecioğlu, F. Federe Kalman Süzgeci ile İç Mekanda Konum Belirleme. 2018 26th Signal Processing and Communications Applications Conference (SIU), 2018, DOI: 10.1109/SIU.2018.8404427.

- [12] Uyar, C., Dervişođlu, İ., Güzel, M., Özdemir, S., Multi sensor based indoor positioning, 2017 International Conference on Computer Science and Engineering (UBMK), 2017, DOI: 10.1109/UBMK.2017.8093568
- [13] İnce, G., Sekmen, B., İç Mekan Servisleri İçin Etkin Bir Beacon Donanımı, İstanbul Teknik Üniversitesi, Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Anabilim Dalı, İstanbul, 2016, 3
- [14] Özcan, C., Saray, F., Tarı, M., Mobil Cihazlar İçin RFID&Bluetooth Düşük Enerji Teknolojisi İle Öğrenci Yoklama Sistemi Tasarımı, International Journal of Multidisciplinary Studies and Innovative Technologies, Konya, 2018, 26-30
- [15] Chintalapudi, K., Iyer, A.P., Padmanabhan, V.N., Indoor localization without the pain, MobiCom '10 Proceedings of the sixteenth annual international conference on Mobile computing and networking, Chicago, 2010, doi 10.1145/1859995.1860016
- [16] Ladd, A.M., Bekris, K.E., Rudys, A.P., Wallach, D.S., Kavraki, L.E., On the feasibility of using wireless ethernet for indoor localization, IEEE Transactions on Robotics and Automation, 2004, Cilt:20(3), DOI: 10.1109/TRA.2004.824948

ÖZGEÇMİŞ

Adı Soyadı : Selahattin Alpcan AĞAYA

Doğum Yeri ve Yılı : Muğla, 1992

Medeni Hali : Evli

Yabancı Dili : İngilizce

E-posta : alpcan_agaya@msn.com

Eğitim Durumu

Lise : Muğla Ortaca Lisesi, 2006.

Lisans :Manisa Celal Bayar Üniversitesi, Elektrik-Elektronik Mühendisliği Bölümü, 2015

Mesleki Deneyim

Tirsan Kardan A.Ş.	2012-2013
EN- KO Elek. Kon. Sis. San. Tic. Ltd. Şti.	2013-2013
Sanal Santral Telekomünikasyon A.Ş.	2014-2015
Likit Telekom & Bilişim	2014-2015
İnan Çelik Kalıp Montaj San. Tic. Ltd. Şti.	2016-2019
WRS Robotik İth. İhr. San. Ve Tic. Ltd. Şti.	2019-Halen

EKLER

Raspberry Pi 3 B+ Python Derleyicisi Kodları

```
# Gelistirici:S. Alpcan AGAYA
# Tarih:24/08/2019 22:50

import blescan
import sys
import requests
import time

import bluetooth._bluetooth as bluez

dev_id = 0
try:
    sock = bluez.hci_open_dev(dev_id)
    print "---iBeacon Tarama Basladi!----"

except:
    print "Bluetooth Cihaz Erisim Hatasi!!!!"
    sys.exit(1)
cihaz="1"
esik=-40
sys.stdout.write("\x1b]2;CIHAZ:"+ cihaz +"\x07")
blescan.hci_le_set_scan_parameters(sock)
blescan.hci_enable_le_scan(sock)
durum1=False
durum1eski=False
sayac1=0
toplaml=0
durum2=False
durum2eski=False
sayac2=0
toplam2=0
durum3=False
durum3eski=False
sayac3=0
toplam3=0
while True:
    durum1=False
    durum2=False
    durum3=False
    sayac1=0
    toplaml=0
    sayac2=0
    toplam2=0
    sayac3=0
    toplam3=0
```

```

returnedList = blescan.parse_events(sock, 50)
print "-----"
time.sleep(1)
for beacon in returnedList:
    mylist = beacon.split(",")
    if mylist[2] == "4660" and mylist[3] == "64001":
        if mylist[0]=="f4:84:4c:4d:dc:50":
            sayac1=sayac1+1
            toplam1=toplam1+int(mylist[5])
        if mylist[0]=="f0:45:da:f6:5b:51":
            sayac2=sayac2+1
            toplam2=toplam2+int(mylist[5])
        if mylist[0]=="4c:96:0e:cb:b1:a7":
            sayac3=sayac3+1
            toplam3=toplam3+int(mylist[5])

    if sayac1>0:
        rssi=toplam1/sayac1
        if rssi>esik:
            durum1=True
            parametreler = {'CIHAZ': cihaz, 'MINOR': "4660", 'MAYOR': "64001",
'RSSI': rssi, 'ETIKET_ID': "501" }
            r = requests.get('http://192.168.10.101/sinyal.asp', params=parametreler)
            print(r.text)
        if sayac2>0:
            rssi=toplam2/sayac2
            if rssi>esik:
                durum2=True
                parametreler = {'CIHAZ': cihaz, 'MINOR': "4660", 'MAYOR': "64001",
'RSSI': rssi, 'ETIKET_ID': "502" }
                r = requests.get('http://192.168.10.101/sinyal.asp', params=parametreler)
                print(r.text)
        if sayac3>0:
            rssi=toplam3/sayac3
            if rssi>esik:
                durum3=True
                parametreler = {'CIHAZ': cihaz, 'MINOR': "4660", 'MAYOR': "64001",
'RSSI': rssi, 'ETIKET_ID': "503" }
                r = requests.get('http://192.168.10.101/sinyal.asp', params=parametreler)
                print(r.text)

    if durum1!=durum1 eski:
        if durum1==True:
            print("Etiket-1 Geldi!")
            durum1 eski=durum1
            etiket_id = "501"

    durum = 1
    payload = {'DURUM': str(durum), 'CIHAZ': cihaz, 'ETIKET_ID': etiket_id}
    r = requests.get('http://192.168.10.101/zone.asp', params=payload)

```

```

        #print(r.text)
    else:
        print("Etiket-1 Gitti!")
        durum1eski=durum1
        etiket_id = "501"

        durum = 0
        payload = {'DURUM': str(durum), 'CIHAZ': cihaz, 'ETIKET_ID': etiket_id}
        r = requests.get('http://192.168.10.101/zone.asp', params=payload)
        #print(r.text)
    if durum2!=durum2eski:
        if durum2==True:
            print("Etiket-2 Geldi!")
            durum2eski=durum2
            etiket_id = "502"

            durum = 1
            payload = {'DURUM': str(durum), 'CIHAZ': cihaz, 'ETIKET_ID': etiket_id}
            r = requests.get('http://192.168.10.101/zone.asp', params=payload)
            #print(r.text)
        else:
            print("Etiket-2 Gitti!")
            etiket_id = "502"

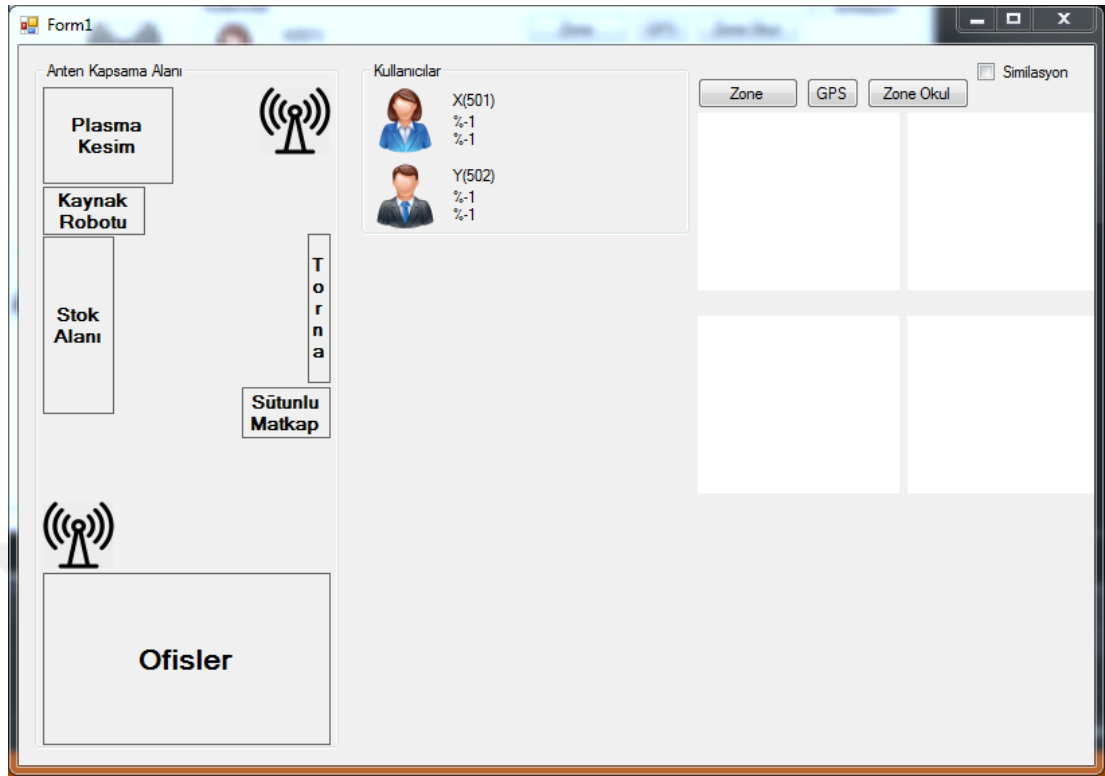
            durum = 0
            payload = {'DURUM': str(durum), 'CIHAZ': cihaz, 'ETIKET_ID': etiket_id}
            r = requests.get('http://192.168.10.101/zone.asp', params=payload)
            durum2eski=durum2
    if durum3!=durum3eski:
        if durum3==True:
            print("Etiket-3 Geldi!")
            durum3eski=durum3
            etiket_id = "503"

            durum = 1
            payload = {'DURUM': str(durum), 'CIHAZ': cihaz, 'ETIKET_ID': etiket_id}
            r = requests.get('http://192.168.10.101/zone.asp', params=payload)
            #print(r.text)
        else:
            print("Etiket-3 Gitti!")
            etiket_id = "503"

            durum = 0
            payload = {'DURUM': str(durum), 'CIHAZ': cihaz, 'ETIKET_ID': etiket_id}
            r = requests.get('http://192.168.10.101/zone.asp', params=payload)
            durum3eski=durum3

```

Visual Studio Arayüz Tasarımı ve Kodları Form-1



```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Data.SqlClient;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Threading.Tasks;  
using System.Windows.Forms;
```

```
namespace YLTEZ
```

```
{  
    public partial class Form1 : Form  
    {
```

```
        int ayserssi1 = 0, alpcanrssi1 = 0, ayserssi2 = 0, alpcanrssi2 = 0, alpcanx = 0,  
        alpcany = 0, aysex = 0, aysey = 0;
```

```
        private void Form1_Load(object sender, EventArgs e)  
        {  
  
        }  
    }
```

```
        private void label7_Click(object sender, EventArgs e)  
        {
```



```

}

private void button1_Click_1(object sender, EventArgs e)
{
    Form2 f2 = new Form2();
    f2.Show();
}

private void button2_Click(object sender, EventArgs e)
{
    Form3 f2 = new Form3();
    f2.Show();
}

private void button3_Click(object sender, EventArgs e)
{
    Form4 f2 = new Form4();
    f2.Show();
}

private void checkBox1_CheckedChanged(object sender, EventArgs e)
{
}

private void checkBox1_CheckStateChanged(object sender, EventArgs e)
{
    timer2.Enabled = checkBox1.Checked;
}

private void timer2_Tick(object sender, EventArgs e)
{
    Random rastgele = new Random();
    int d123c1 = rastgele.Next(100);
    int d123c2 = rastgele.Next(100);
    int d103c1 = rastgele.Next(100);
    int d103c2 = rastgele.Next(100);

```

```

webBrowser1.Navigate("http://127.0.0.1/sinyal.asp?ETIKET_ID=123&RSSI=-
+d123c1+"&MAYOR=1445&MINOR=55868&CIHAZ=1");

```

```

webBrowser2.Navigate("http://127.0.0.1/sinyal.asp?ETIKET_ID=123&RSSI=-
+d123c2+"&MAYOR=1445&MINOR=55868&CIHAZ=2");

```

```

webBrowser3.Navigate("http://127.0.0.1/sinyal.asp?ETIKET_ID=103&RSSI=-
+d103c1+"&MAYOR=1445&MINOR=55868&CIHAZ=1");

```

```

webBrowser4.Navigate("http://127.0.0.1/sinyal.asp?ETIKET_ID=103&RSSI=-
+d103c2+"&MAYOR=1445&MINOR=55868&CIHAZ=2");
}

private void timer1_Tick(object sender, EventArgs e)
{
    SQLOKU();
}

public Form1()
{
    InitializeComponent();
}

private void SQLOKU()
{
    SqlConnection baglanti = new SqlConnection();
    baglanti.ConnectionString = @"Data Source=;Initial Catalog=;User Id=;
password=";
    SqlCommand komut = new SqlCommand();
    komut.CommandText = "SELECT * FROM CANLI_IZLE";
    komut.Connection = baglanti;
    komut.CommandType = CommandType.Text;

    SqlDataReader dr;
    baglanti.Open();
    dr = komut.ExecuteReader();
    ayserssi1 = 1;
    ayserssi2 = 1;
    alpcanrssi1 = 1;
    alpcanrssi2 = 1;
    while (dr.Read())
    {
        if (dr["ETIKET_ID"].ToString() == "502" && dr["CIHAZ"].ToString()
== "1")
            ayserssi1 = Convert.ToInt16(dr["RSSI"]);
        if (dr["ETIKET_ID"].ToString() == "502" && dr["CIHAZ"].ToString()
== "2")
            ayserssi2 = Convert.ToInt16(dr["RSSI"]);
        if (dr["ETIKET_ID"].ToString() == "501" && dr["CIHAZ"].ToString()
== "1")
            alpcanrssi1 = Convert.ToInt16(dr["RSSI"]);
        if (dr["ETIKET_ID"].ToString() == "501" && dr["CIHAZ"].ToString()
== "2")
            alpcanrssi2 = Convert.ToInt16(dr["RSSI"]);
    }
    baglanti.Close();
    ayserssi1 *= -1;
    ayserssi2 *= -1;
}

```

```

alpcanrssi1 *= -1;
alpcanrssi2 *= -1;

aysey = 75+(ayserssi1 * 161 / (40));
aysex = 6+(ayserssi2 * 196 / (40));
alpcany = 75+(alpcanrssi1 * 161 / (40));
alpcanx = 6+(alpcanrssi2 * 196 / (40)) ;

//MessageBox.Show("ayse:"+ayserssi1.ToString() + "/" +
ayserssi2.ToString()+"/"+ aysex.ToString()+"/"+aysey.ToString());
//MessageBox.Show("alpcan:"+alpcanrssi1.ToString() + "/" +
alpcanrssi2.ToString() + "/" + alpcanx.ToString() + "/" + alpcanx.ToString());
label3.Text = "%" + ayserssi1.ToString();
label5.Text = "%" + ayserssi2.ToString();
label4.Text = "%" + alpcanrssi1.ToString();
label6.Text = "%" + alpcanrssi2.ToString();

aysegez.Location = new Point(aysex, aysey);
alpcangez.Location = new Point(alpcanx, alpcany);
if (ayserssi1 == -1 && ayserssi2 == -1)
    aysegez.Visible = false;
else
    aysegez.Visible = true;

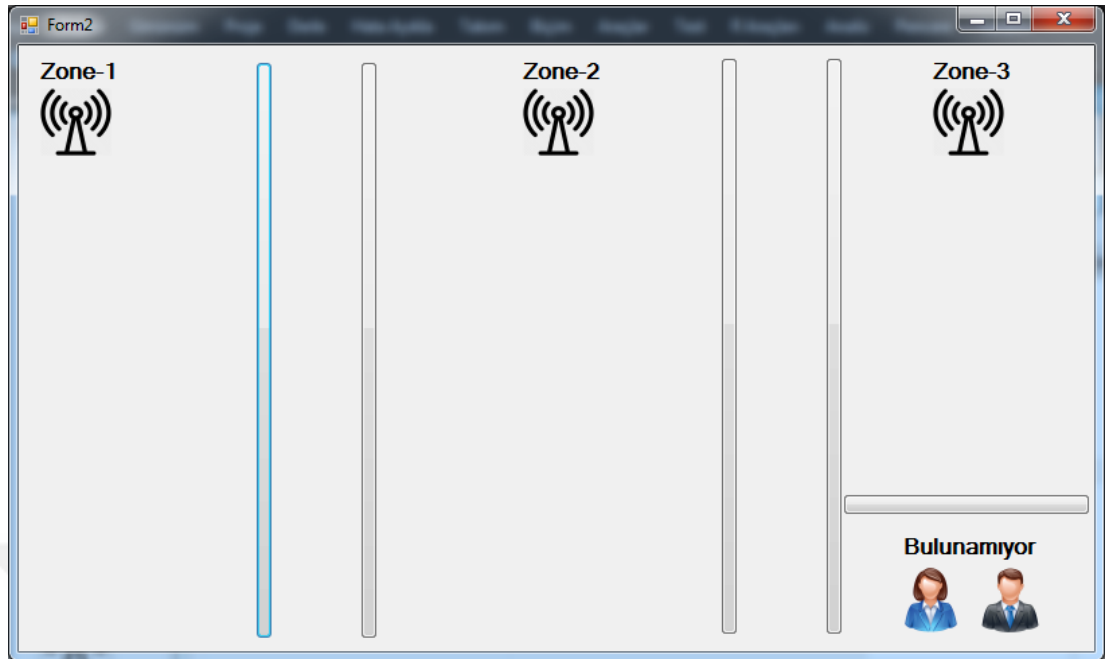
if (alpcanrssi1 == -1 && alpcanrssi2 == -1)
    alpcangez.Visible = false;
else
    alpcangez.Visible = true;

}

private void button1_Click(object sender, EventArgs e)
{
    SQLOKU();
}
}
}

```

Visual Studio Arayüz Tasarımı ve Kodları Form-2



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
```

```
namespace YLTEZ
{
    public partial class Form2 : Form
    {
        public Form2()
        {
            InitializeComponent();
        }
        int alpcanx = 0, alpcany = 0, aysex = 0, aysey = 0;

        private void Form2_Load(object sender, EventArgs e)
        {
            timer1.Enabled = true;
        }

        private void timer1_Tick(object sender, EventArgs e)
        {
            SQLOKU();
        }
    }
}
```

```

    }

    private void SLOKUKU()
    {
        SqlConnection baglanti = new SqlConnection();
        baglanti.ConnectionString = @"Data Source=;Initial Catalog=;User Id=;
password=";
        SqlCommand komut = new SqlCommand();
        komut.CommandText = "select ETIKET_ID, sum(ZONEM1) as ZONEM1,
sum(ZONEM2) as ZONEM2, sum(ZONEM3) as ZONEM3 from ZONE_CANLI
group by ETIKET_ID";
        komut.Connection = baglanti;
        komut.CommandType = CommandType.Text;
        Random rastgele = new Random();
        SqlDataReader dr;
        baglanti.Open();
        dr = komut.ExecuteReader();
        bool eti1 = false, eti2 = false;
        while (dr.Read())
        {
            if (dr["ETIKET_ID"].ToString() == "502")
                eti2 = true;
            if (dr["ETIKET_ID"].ToString() == "502" && dr["ZONEM1"].ToString()
== "1" && dr["ZONEM2"].ToString() == "1")
                pictureBox5.Location = new Point(195, rastgele.Next(13,391));
            if (dr["ETIKET_ID"].ToString() == "502" && dr["ZONEM1"].ToString()
== "0" && dr["ZONEM2"].ToString() == "1")
                pictureBox5.Location = new Point(rastgele.Next(273, 463),
rastgele.Next(92, 391));
            if (dr["ETIKET_ID"].ToString() == "502" && dr["ZONEM1"].ToString()
== "1" && dr["ZONEM2"].ToString() == "0")
                pictureBox5.Location = new Point(rastgele.Next(12, 117),
rastgele.Next(92, 391));
            if (dr["ETIKET_ID"].ToString() == "502" && dr["ZONEM2"].ToString()
== "1" && dr["ZONEM3"].ToString() == "1")
                pictureBox5.Location = new Point(541, rastgele.Next(92, 391));
            if (dr["ETIKET_ID"].ToString() == "502" && dr["ZONEM2"].ToString()
== "0" && dr["ZONEM3"].ToString() == "1")
                pictureBox5.Location = new Point(rastgele.Next(619, 735),
rastgele.Next(92, 287));

            if (dr["ETIKET_ID"].ToString() == "501")
                eti1 = true;
            if (dr["ETIKET_ID"].ToString() == "501" && dr["ZONEM1"].ToString()
== "1" && dr["ZONEM2"].ToString() == "1")
                pictureBox4.Location = new Point(195, rastgele.Next(13, 391));

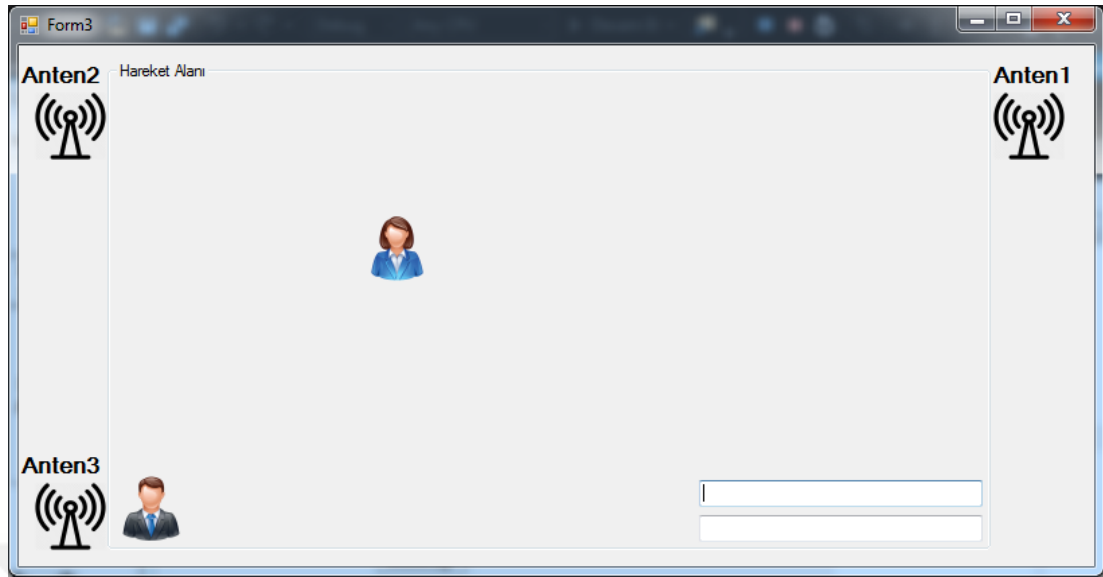
```

```

        if (dr["ETIKET_ID"].ToString() == "501" && dr["ZONEM1"].ToString()
== "0" && dr["ZONEM2"].ToString() == "1")
            pictureBox4.Location = new Point(rastgele.Next(273, 463),
rastgele.Next(92, 391));
        if (dr["ETIKET_ID"].ToString() == "501" && dr["ZONEM1"].ToString()
== "1" && dr["ZONEM2"].ToString() == "0")
            pictureBox4.Location = new Point(rastgele.Next(12, 117),
rastgele.Next(92, 391));
        if (dr["ETIKET_ID"].ToString() == "501" && dr["ZONEM2"].ToString()
== "1" && dr["ZONEM3"].ToString() == "1")
            pictureBox4.Location = new Point(541, rastgele.Next(92, 391));
        if (dr["ETIKET_ID"].ToString() == "501" && dr["ZONEM2"].ToString()
== "0" && dr["ZONEM3"].ToString() == "1")
            pictureBox4.Location = new Point(rastgele.Next(619, 735),
rastgele.Next(92, 287));
    }
    if (!leti2)
        pictureBox5.Location = new Point(652, 387);
    if (!leti1)
        pictureBox4.Location = new Point(711, 387);
        baglanti.Close();
    }
}
}

```

Visual Studio Arayüz Tasarımı ve Kodları Form-3



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
```

```
namespace YLTEZ
```

```
{
    public partial class Form3 : Form
    {
        public Form3()
        {
            InitializeComponent();
        }
        int[] rssi501={ 0,0,0}, rssi502= { 0,0,0};
        private void Form3_Load(object sender, EventArgs e)
        {
            timer1.Enabled = true;
        }

        private void SQLOKU()
        {
            SqlConnection baglanti = new SqlConnection();
            baglanti.ConnectionString = @"Data Source= ;Initial Catalog=;User Id=;
password=";
            SqlCommand komut = new SqlCommand();
```

```

        komut.CommandText = "select ETIKET_ID,ISNULL(RSSI1,0) as
RSSI1,ISNULL(RSSI2,0) as RSSI2,ISNULL(RSSI3,0) as RSSI3 from
CANLI_GPS";
        komut.Connection = baglanti;
        komut.CommandType = CommandType.Text;
        SqlDataReader dr;
        baglanti.Open();
        dr = komut.ExecuteReader();
        rssi501[0] = 0;
        rssi501[1] = 0;
        rssi501[2] = 0;
        rssi502[0] = 0;
        rssi502[1] = 0;
        rssi502[2] = 0;
        while (dr.Read())
        {
            if (dr["ETIKET_ID"].ToString() == "501")
            {
                rssi501[0] = -1 * Convert.ToInt16(dr["RSSI1"].ToString());
                rssi501[1] = -1 * Convert.ToInt16(dr["RSSI2"].ToString());
                rssi501[2] = -1 * Convert.ToInt16(dr["RSSI3"].ToString());
            }
            if (dr["ETIKET_ID"].ToString() == "502")
            {
                rssi502[0] = -1 * Convert.ToInt16(dr["RSSI1"].ToString());
                rssi502[1] = -1 * Convert.ToInt16(dr["RSSI2"].ToString());
                rssi502[2] = -1 * Convert.ToInt16(dr["RSSI3"].ToString());
            }
            textBox1.Text = "501 |" + rssi501[0].ToString() + "|" +
rssi501[1].ToString() + "|" + rssi501[2].ToString();
            textBox2.Text = "502 |" + rssi502[0].ToString() + "|" +
rssi502[1].ToString() + "|" + rssi502[2].ToString();
            if (rssi501[0] != 0 && rssi501[1] != 0 && rssi501[2] != 0)
                e501p.Location = new Point(19 + (rssi501[1] * ((596-6) / (rssi501[1] +
rssi501[0]))), 6 + (rssi501[1] * ((307-19) / (rssi501[1] + rssi501[2]))));
            if (rssi502[0] != 0 && rssi502[1] != 0 && rssi502[2] != 0)
                e502p.Location = new Point(19 + (rssi502[1] * ((596-6) / (rssi502[1] +
rssi502[0]))), 6 + (rssi502[1] * ((307-19) / (rssi502[1] + rssi502[2]))));
        }
        baglanti.Close();
    }

    private void timer1_Tick(object sender, EventArgs e)
    {
        SQLOKUKU();
    }

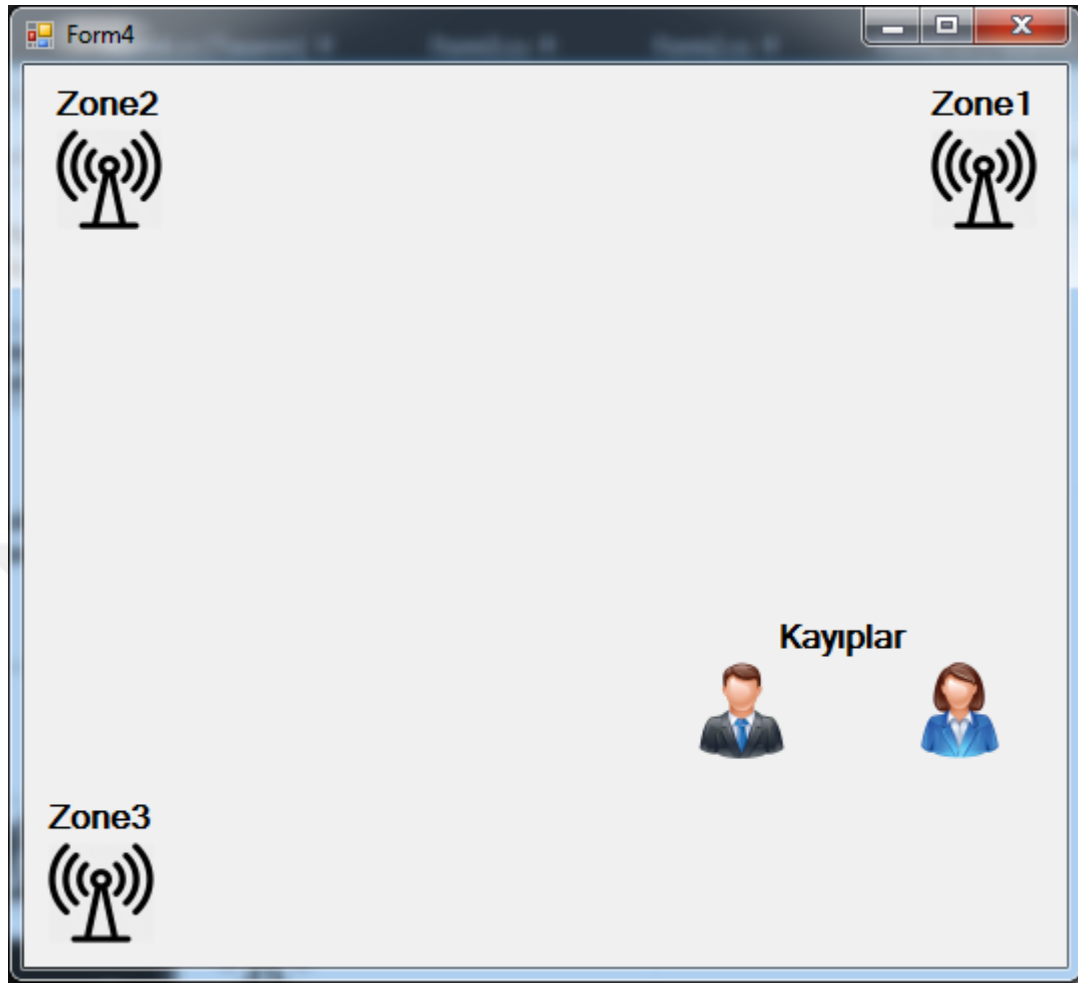
    private void label3_Click(object sender, EventArgs e)
    {

```


}
}
}



Visual Studio Arayüz Tasarımı ve Kodları Form-4



```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.SqlClient;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace YLTEZ
{
    public partial class Form4 : Form
    {
        public Form4()
        {
            InitializeComponent();
        }
        private void SQLOKU()
        {
```

```

SqlConnection baglanti = new SqlConnection();
baglanti.ConnectionString = @"Data Source=COMPUTER\WINCC;Initial
Catalog=BEACON;User Id=alpcanagaya; password=171830";
SqlCommand komut = new SqlCommand();
komut.CommandText = "select ETIKET_ID, sum(ZONEM1) as ZONEM1,
sum(ZONEM2) as ZONEM2, sum(ZONEM3) as ZONEM3 from ZONE_CANLI
group by ETIKET_ID";
komut.Connection = baglanti;
komut.CommandType = CommandType.Text;
Random rastgele = new Random();
SqlDataReader dr;
baglanti.Open();
dr = komut.ExecuteReader();
bool eti1 = false, eti2 = false;
while (dr.Read())
{
    if (dr["ETIKET_ID"].ToString() == "502")
        eti2 = true;
    if (dr["ETIKET_ID"].ToString() == "502" && dr["ZONEM1"].ToString()
== "1" && dr["ZONEM2"].ToString() == "1")
        e502p.Location = new Point(230, rastgele.Next(12, 125));
    if (dr["ETIKET_ID"].ToString() == "502" && dr["ZONEM1"].ToString()
== "0" && dr["ZONEM2"].ToString() == "1")
        e502p.Location = new Point(rastgele.Next(16, 155), rastgele.Next(32,
125));
    if (dr["ETIKET_ID"].ToString() == "502" && dr["ZONEM1"].ToString()
== "1" && dr["ZONEM2"].ToString() == "0")
        e502p.Location = new Point(rastgele.Next(305, 453), rastgele.Next(12,
125));
    if (dr["ETIKET_ID"].ToString() == "502" && dr["ZONEM2"].ToString()
== "1" && dr["ZONEM3"].ToString() == "1")
        e502p.Location = new Point(rastgele.Next(12, 155), rastgele.Next(197,
206));
    if (dr["ETIKET_ID"].ToString() == "502" && dr["ZONEM2"].ToString()
== "0" && dr["ZONEM3"].ToString() == "1")
        e502p.Location = new Point(rastgele.Next(12, 155), rastgele.Next(278,
388));

    if (dr["ETIKET_ID"].ToString() == "501")
        eti1 = true;
    if (dr["ETIKET_ID"].ToString() == "501" && dr["ZONEM1"].ToString()
== "1" && dr["ZONEM2"].ToString() == "1")
        e501p.Location = new Point(230, rastgele.Next(12, 125));
    if (dr["ETIKET_ID"].ToString() == "501" && dr["ZONEM1"].ToString()
== "0" && dr["ZONEM2"].ToString() == "1")
        e501p.Location = new Point(rastgele.Next(16, 155), rastgele.Next(32,
125));
}

```

```

        if (dr["ETIKET_ID"].ToString() == "501" && dr["ZONEM1"].ToString()
== "1" && dr["ZONEM2"].ToString() == "0")
            e501p.Location = new Point(rastgele.Next(305, 453), rastgele.Next(12,
125));
        if (dr["ETIKET_ID"].ToString() == "501" && dr["ZONEM2"].ToString()
== "1" && dr["ZONEM3"].ToString() == "1")
            e501p.Location = new Point(rastgele.Next(12, 155), rastgele.Next(197,
206));
        if (dr["ETIKET_ID"].ToString() == "501" && dr["ZONEM2"].ToString()
== "0" && dr["ZONEM3"].ToString() == "1")
            e501p.Location = new Point(rastgele.Next(12, 155), rastgele.Next(278,
388));
    }
    if (!eti2)
        e502p.Location = new Point(441, 297);
    if (!eti1)
        e501p.Location = new Point(332, 297);
    baglanti.Close();
}
private void timer1_Tick(object sender, EventArgs e)
{
    SQLOKU();
}

private void Form4_Load(object sender, EventArgs e)
{
    timer1.Enabled = true;
}
}
}
}

```

MsSQL Express Veritabanı Şema Scripti

```
USE [BEACON]
GO
/***** Object: Table [dbo].[PERSONEL]    Script Date: 28.08.2019 10:52:57
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[PERSONEL](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [ADI_SOYADI] [nvarchar](max) NULL,
    [CEP] [nvarchar](max) NULL,
    [ETIKET_ID] [nvarchar](max) NULL,
    CONSTRAINT [PK_PERSONEL] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
/***** Object: Table [dbo].[SINYALLER]    Script Date: 28.08.2019 10:52:57
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[SINYALLER](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [ETIKET_ID] [nvarchar](max) NOT NULL,
    [RSSI] [int] NOT NULL,
    [MAYOR] [int] NOT NULL,
    [MINOR] [int] NOT NULL,
    [CIHAZ] [int] NOT NULL,
    [TARIH] [datetime] NOT NULL,
    CONSTRAINT [PK_SINYALLER] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
/***** Object: Table [dbo].[ZONE_KAYITLAR]    Script Date: 28.08.2019
10:52:57 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
```

```

GO
CREATE TABLE [dbo].[ZONE_KAYITLAR](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [ETIKET_ID] [int] NOT NULL,
    [CIHAZ_ID] [int] NOT NULL,
    [GIRIS_ZAMAN] [datetime] NOT NULL,
    [CIKIS_ZAMAN] [datetime] NULL,
    CONSTRAINT [PK_ZONELER] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY]
GO
/***** Object: Table [dbo].[ZONE_TANIM]    Script Date: 28.08.2019 10:52:57
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[ZONE_TANIM](
    [ID] [int] IDENTITY(1,1) NOT NULL,
    [ZONE_ADI] [nvarchar](max) NULL,
    [ZONE_ACIKLAMA] [nvarchar](max) NULL,
    [ZONE_CIHAZ] [nvarchar](max) NULL,
    CONSTRAINT [PK_ZONE_TANIM] PRIMARY KEY CLUSTERED
(
    [ID] ASC
)WITH (PAD_INDEX = OFF, STATISTICS_NORECOMPUTE = OFF,
IGNORE_DUP_KEY = OFF, ALLOW_ROW_LOCKS = ON,
ALLOW_PAGE_LOCKS = ON) ON [PRIMARY]
) ON [PRIMARY] TEXTIMAGE_ON [PRIMARY]
GO
/***** Object: View [dbo].[CANLI_IZLE]    Script Date: 28.08.2019 10:52:57
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE VIEW [dbo].[CANLI_IZLE]
AS
SELECT      *
FROM        SINYALLER
WHERE       ID =
            (SELECT      max(ID) AS ID
             FROM        SINYALLER
             WHERE       ETIKET_ID = 503 AND CIHAZ = 1
             GROUP BY   ETIKET_ID) AND TARİH >= DATEADD(second,
- 10, GETDATE())

```

```

UNION ALL
SELECT      *
FROM        SINYALLER
WHERE       ID =
            (SELECT      max(ID) AS ID
             FROM        SINYALLER
             WHERE       ETIKET_ID = 502 AND CIHAZ = 1
             GROUP BY   ETIKET_ID) AND TARIH >= DATEADD(second,
- 10, GETDATE())
UNION ALL
SELECT      *
FROM        SINYALLER
WHERE       ID =
            (SELECT      max(ID) AS ID
             FROM        SINYALLER
             WHERE       ETIKET_ID = 501 AND CIHAZ = 1
             GROUP BY   ETIKET_ID) AND TARIH >= DATEADD(second,
- 10, GETDATE())
UNION ALL
SELECT      *
FROM        SINYALLER
WHERE       ID =
            (SELECT      max(ID) AS ID
             FROM        SINYALLER
             WHERE       ETIKET_ID = 503 AND CIHAZ = 2
             GROUP BY   ETIKET_ID) AND TARIH >= DATEADD(second,
- 10, GETDATE())
UNION ALL
SELECT      *
FROM        SINYALLER
WHERE       ID =
            (SELECT      max(ID) AS ID
             FROM        SINYALLER
             WHERE       ETIKET_ID = 502 AND CIHAZ = 2
             GROUP BY   ETIKET_ID) AND TARIH >= DATEADD(second,
- 10, GETDATE())
UNION ALL
SELECT      *
FROM        SINYALLER
WHERE       ID =
            (SELECT      max(ID) AS ID
             FROM        SINYALLER
             WHERE       ETIKET_ID = 501 AND CIHAZ = 2
             GROUP BY   ETIKET_ID) AND TARIH >= DATEADD(second,
- 10, GETDATE())
UNION ALL
SELECT      *
FROM        SINYALLER
WHERE       ID =
            (SELECT      max(ID) AS ID

```

```

FROM SINYALLER
WHERE ETIKET_ID = 503 AND CIHAZ = 3
GROUP BY ETIKET_ID) AND TARİH >= DATEADD(second,
- 10, GETDATE())
UNION ALL
SELECT *
FROM SINYALLER
WHERE ID =
(SELECT max(ID) AS ID
FROM SINYALLER
WHERE ETIKET_ID = 502 AND CIHAZ = 3
GROUP BY ETIKET_ID) AND TARİH >= DATEADD(second,
- 10, GETDATE())
UNION ALL
SELECT *
FROM SINYALLER
WHERE ID =
(SELECT max(ID) AS ID
FROM SINYALLER
WHERE ETIKET_ID = 501 AND CIHAZ = 3
GROUP BY ETIKET_ID) AND TARİH >= DATEADD(second,
- 10, GETDATE())
GO
/***** Object: View [dbo].[CANLI1] Script Date: 28.08.2019 10:52:57 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE VIEW [dbo].[CANLI1]
AS
SELECT DISTINCT ETIKET_ID
FROM dbo.CANLI_IZLE
GO
/***** Object: View [dbo].[CANLI_GPS] Script Date: 28.08.2019 10:52:57
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE VIEW [dbo].[CANLI_GPS]
AS
SELECT ETIKET_ID,
(SELECT RSSI
FROM dbo.CANLI_IZLE
WHERE (ETIKET_ID = c1.ETIKET_ID) AND (CIHAZ = 1))
AS RSSI1,
(SELECT RSSI
FROM dbo.CANLI_IZLE AS CANLI_IZLE_2
WHERE (ETIKET_ID = c1.ETIKET_ID) AND (CIHAZ = 2))
AS RSSI2,

```



```

                (SELECT      RSSI
                 FROM        dbo.CANLI_IZLE AS CANLI_IZLE_1
                 WHERE      (ETIKET_ID = c1.ETIKET_ID) AND (CIHAZ = 3))
AS RSSI3
FROM      dbo.CANLI1 AS c1
GO
/***** Object: View [dbo].[ZONE_GIRIS_CIKIS]      Script Date: 28.08.2019
10:52:57 *****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE VIEW [dbo].[ZONE_GIRIS_CIKIS]
AS
SELECT      ETIKET_ID, CIHAZ_ID, GIRIS_ZAMAN, CONVERT(time,
CIKIS_ZAMAN - GIRIS_ZAMAN) AS SURE
FROM        dbo.ZONE_KAYITLAR
GO
/***** Object: View [dbo].[ZONE_SURE]      Script Date: 28.08.2019 10:52:57
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE VIEW [dbo].[ZONE_SURE]
AS
SELECT      ETIKET_ID, CIHAZ_ID, SUM((DATEPART(SECOND, SURE) + 60
* DATEPART(MINUTE, SURE)) + 3600 * DATEPART(HOUR, SURE)) AS SURE
FROM        dbo.ZONE_GIRIS_CIKIS
WHERE      (GIRIS_ZAMAN BETWEEN DATEADD(day, - 100, GETDATE())
AND GETDATE())
GROUP BY ETIKET_ID, CIHAZ_ID
GO
/***** Object: View [dbo].[ZONE_CANLI]      Script Date: 28.08.2019 10:52:57
*****/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE VIEW [dbo].[ZONE_CANLI]
AS
SELECT      ETIKET_ID, (CASE CIHAZ_ID WHEN '1' THEN 1 ELSE 0 END) AS
ZONEM1, (CASE CIHAZ_ID WHEN '2' THEN 1 ELSE 0 END) AS ZONEM2,
(CASE CIHAZ_ID WHEN '3' THEN 1 ELSE 0 END) AS ZONEM3
FROM        dbo.ZONE_KAYITLAR
WHERE      (CIKIS_ZAMAN IS NULL)
GO
/***** Object: View [dbo].[ZONE_RAPOR_VIEW]      Script Date: 28.08.2019
10:52:57 *****/
SET ANSI_NULLS ON

```

```

GO
SET QUOTED_IDENTIFIER ON
GO
CREATE VIEW [dbo].[ZONE_RAPOR_VIEW]
AS
SELECT          P.ADI_SOYADI, ZT.ZONE_ADI, ZK.GIRIS_ZAMAN,
ZK.CIKIS_ZAMAN, CONVERT(time, ZK.CIKIS_ZAMAN - ZK.GIRIS_ZAMAN)
AS SURE
FROM          dbo.ZONE_KAYITLAR AS ZK LEFT OUTER JOIN
              dbo.ZONE_TANIM AS ZT ON ZK.ZONE_ID = ZT.ID LEFT
OUTER JOIN
              dbo.PERSONEL AS P ON ZK.ETIKET_ID = P.ETIKET_ID

GO
EXEC          sys.sp_addextendedproperty          @name=N'MS_DiagramPane1',
@value=N'[0E232FF0-B466-11cf-A24F-00AA00A3EFFF, 1.00]
Begin DesignProperties =
  Begin PaneConfigurations =
    Begin PaneConfiguration = 0
      NumPanes = 4
      Configuration = "(H (1[40] 4[20] 2[20] 3) )"
    End
    Begin PaneConfiguration = 1
      NumPanes = 3
      Configuration = "(H (1 [50] 4 [25] 3))"
    End
    Begin PaneConfiguration = 2
      NumPanes = 3
      Configuration = "(H (1 [50] 2 [25] 3))"
    End
    Begin PaneConfiguration = 3
      NumPanes = 3
      Configuration = "(H (4 [30] 2 [40] 3))"
    End
    Begin PaneConfiguration = 4
      NumPanes = 2
      Configuration = "(H (1 [56] 3))"
    End
    Begin PaneConfiguration = 5
      NumPanes = 2
      Configuration = "(H (2 [66] 3))"
    End
    Begin PaneConfiguration = 6
      NumPanes = 2
      Configuration = "(H (4 [50] 3))"
    End
    Begin PaneConfiguration = 7
      NumPanes = 1
      Configuration = "(V (3))"
    End
    Begin PaneConfiguration = 8

```

```

    NumPanels = 3
    Configuration = "(H (1[56] 4[18] 2) )"
End
Begin PaneConfiguration = 9
    NumPanels = 2
    Configuration = "(H (1 [75] 4))"
End
Begin PaneConfiguration = 10
    NumPanels = 2
    Configuration = "(H (1[66] 2) )"
End
Begin PaneConfiguration = 11
    NumPanels = 2
    Configuration = "(H (4 [60] 2))"
End
Begin PaneConfiguration = 12
    NumPanels = 1
    Configuration = "(H (1) )"
End
Begin PaneConfiguration = 13
    NumPanels = 1
    Configuration = "(V (4))"
End
Begin PaneConfiguration = 14
    NumPanels = 1
    Configuration = "(V (2))"
End
ActivePaneConfig = 0
End
Begin DiagramPane =
    Begin Origin =
        Top = 0
        Left = 0
    End
    Begin Tables =
        Begin Table = "c1"
            Begin Extent =
                Top = 6
                Left = 38
                Bottom = 84
                Right = 208
            End
            DisplayFlags = 280
            TopColumn = 0
        End
    End
End
Begin SQLPane =
End
Begin DataPane =

```

```

Begin ParameterDefaults = ""
End
End
Begin CriteriaPane =
Begin ColumnWidths = 11
Column = 1440
Alias = 900
Table = 1170
Output = 720
Append = 1400
NewValue = 1170
SortType = 1350
SortOrder = 1410
GroupBy = 1350
Filter = 1350
Or = 1350
Or = 1350
Or = 1350
End
End
End
',
@level0type=N'SCHEMA',@level0name=N'dbo',
@level1type=N'VIEW',@level1name=N'CANLI_GPS'
GO
EXEC sys.sp_addextendedproperty @name=N'MS_DiagramPaneCount', @value=1 ,
@level0type=N'SCHEMA',@level0name=N'dbo',
@level1type=N'VIEW',@level1name=N'CANLI_GPS'
GO
EXEC sys.sp_addextendedproperty @name=N'MS_DiagramPane1',
@value=N'[0E232FF0-B466-11cf-A24F-00AA00A3EFFF, 1.00]
Begin DesignProperties =
Begin PaneConfigurations =
Begin PaneConfiguration = 0
NumPanes = 4
Configuration = "(H (1[40] 4[20] 2[20] 3) )"
End
Begin PaneConfiguration = 1
NumPanes = 3
Configuration = "(H (1 [50] 4 [25] 3))"
End
Begin PaneConfiguration = 2
NumPanes = 3
Configuration = "(H (1 [50] 2 [25] 3))"
End
Begin PaneConfiguration = 3
NumPanes = 3
Configuration = "(H (4 [30] 2 [40] 3))"
End
Begin PaneConfiguration = 4
NumPanes = 2

```

```

    Configuration = "(H (1 [56] 3))"
End
Begin PaneConfiguration = 5
    NumPanels = 2
    Configuration = "(H (2 [66] 3))"
End
Begin PaneConfiguration = 6
    NumPanels = 2
    Configuration = "(H (4 [50] 3))"
End
Begin PaneConfiguration = 7
    NumPanels = 1
    Configuration = "(V (3))"
End
Begin PaneConfiguration = 8
    NumPanels = 3
    Configuration = "(H (1[56] 4[18] 2) )"
End
Begin PaneConfiguration = 9
    NumPanels = 2
    Configuration = "(H (1 [75] 4))"
End
Begin PaneConfiguration = 10
    NumPanels = 2
    Configuration = "(H (1[66] 2) )"
End
Begin PaneConfiguration = 11
    NumPanels = 2
    Configuration = "(H (4 [60] 2))"
End
Begin PaneConfiguration = 12
    NumPanels = 1
    Configuration = "(H (1) )"
End
Begin PaneConfiguration = 13
    NumPanels = 1
    Configuration = "(V (4))"
End
Begin PaneConfiguration = 14
    NumPanels = 1
    Configuration = "(V (2))"
End
ActivePaneConfig = 0
End
Begin DiagramPane =
    Begin Origin =
        Top = 0
        Left = 0
    End
    Begin Tables =

```

```

    End
End
Begin SQLPane =
End
Begin DataPane =
    Begin ParameterDefaults = ""
    End
End
Begin CriteriaPane =
    Begin ColumnWidths = 11
        Column = 1440
        Alias = 900
        Table = 1170
        Output = 720
        Append = 1400
        NewValue = 1170
        SortType = 1350
        SortOrder = 1410
        GroupBy = 1350
        Filter = 1350
        Or = 1350
        Or = 1350
        Or = 1350
    End
End
End
',
@level0type=N'SCHEMA',@level0name=N'dbo',
@level1type=N'VIEW',@level1name=N'CANLI_IZLE'
GO
EXEC sys.sp_addextendedproperty @name=N'MS_DiagramPaneCount', @value=1 ,
@level0type=N'SCHEMA',@level0name=N'dbo',
@level1type=N'VIEW',@level1name=N'CANLI_IZLE'
GO
EXEC sys.sp_addextendedproperty @name=N'MS_DiagramPane1',
@value=N'[0E232FF0-B466-11cf-A24F-00AA00A3EFFF, 1.00]
Begin DesignProperties =
    Begin PaneConfigurations =
        Begin PaneConfiguration = 0
            NumPanes = 4
            Configuration = "(H (1[40] 4[20] 2[20] 3) )"
        End
        Begin PaneConfiguration = 1
            NumPanes = 3
            Configuration = "(H (1 [50] 4 [25] 3))"
        End
        Begin PaneConfiguration = 2
            NumPanes = 3
            Configuration = "(H (1 [50] 2 [25] 3))"
        End
        Begin PaneConfiguration = 3

```

```

    NumPanels = 3
    Configuration = "(H (4 [30] 2 [40] 3))"
End
Begin PaneConfiguration = 4
    NumPanels = 2
    Configuration = "(H (1 [56] 3))"
End
Begin PaneConfiguration = 5
    NumPanels = 2
    Configuration = "(H (2 [66] 3))"
End
Begin PaneConfiguration = 6
    NumPanels = 2
    Configuration = "(H (4 [50] 3))"
End
Begin PaneConfiguration = 7
    NumPanels = 1
    Configuration = "(V (3))"
End
Begin PaneConfiguration = 8
    NumPanels = 3
    Configuration = "(H (1[56] 4[18] 2) )"
End
Begin PaneConfiguration = 9
    NumPanels = 2
    Configuration = "(H (1 [75] 4))"
End
Begin PaneConfiguration = 10
    NumPanels = 2
    Configuration = "(H (1[66] 2) )"
End
Begin PaneConfiguration = 11
    NumPanels = 2
    Configuration = "(H (4 [60] 2))"
End
Begin PaneConfiguration = 12
    NumPanels = 1
    Configuration = "(H (1) )"
End
Begin PaneConfiguration = 13
    NumPanels = 1
    Configuration = "(V (4))"
End
Begin PaneConfiguration = 14
    NumPanels = 1
    Configuration = "(V (2))"
End
ActivePaneConfig = 0
End
Begin DiagramPane =

```

```

Begin Origin =
  Top = 0
  Left = 0
End
Begin Tables =
  Begin Table = "CANLI_IZLE"
    Begin Extent =
      Top = 6
      Left = 38
      Bottom = 135
      Right = 208
    End
    DisplayFlags = 280
    TopColumn = 0
  End
End
End
Begin SQLPane =
End
Begin DataPane =
  Begin ParameterDefaults = ""
  End
End
Begin CriteriaPane =
  Begin ColumnWidths = 11
    Column = 1440
    Alias = 900
    Table = 1170
    Output = 720
    Append = 1400
    NewValue = 1170
    SortType = 1350
    SortOrder = 1410
    GroupBy = 1350
    Filter = 1350
    Or = 1350
    Or = 1350
    Or = 1350
  End
End
End
',
@level0type=N'SCHEMA',@level0name=N'dbo',
@level1type=N'VIEW',@level1name=N'CANLI1'
GO
EXEC sys.sp_addextendedproperty @name=N'MS_DiagramPaneCount', @value=1 ,
@level0type=N'SCHEMA',@level0name=N'dbo',
@level1type=N'VIEW',@level1name=N'CANLI1'
GO
EXEC sys.sp_addextendedproperty @name=N'MS_DiagramPane1',
@value=N'[0E232FF0-B466-11cf-A24F-00AA00A3EFFF, 1.00]

```



```

Begin DesignProperties =
  Begin PaneConfigurations =
    Begin PaneConfiguration = 0
      NumPanels = 4
      Configuration = "(H (1[40] 4[20] 2[20] 3) )"
    End
    Begin PaneConfiguration = 1
      NumPanels = 3
      Configuration = "(H (1 [50] 4 [25] 3))"
    End
    Begin PaneConfiguration = 2
      NumPanels = 3
      Configuration = "(H (1 [50] 2 [25] 3))"
    End
    Begin PaneConfiguration = 3
      NumPanels = 3
      Configuration = "(H (4 [30] 2 [40] 3))"
    End
    Begin PaneConfiguration = 4
      NumPanels = 2
      Configuration = "(H (1 [56] 3))"
    End
    Begin PaneConfiguration = 5
      NumPanels = 2
      Configuration = "(H (2 [66] 3))"
    End
    Begin PaneConfiguration = 6
      NumPanels = 2
      Configuration = "(H (4 [50] 3))"
    End
    Begin PaneConfiguration = 7
      NumPanels = 1
      Configuration = "(V (3))"
    End
    Begin PaneConfiguration = 8
      NumPanels = 3
      Configuration = "(H (1[56] 4[18] 2) )"
    End
    Begin PaneConfiguration = 9
      NumPanels = 2
      Configuration = "(H (1 [75] 4))"
    End
    Begin PaneConfiguration = 10
      NumPanels = 2
      Configuration = "(H (1[66] 2) )"
    End
    Begin PaneConfiguration = 11
      NumPanels = 2
      Configuration = "(H (4 [60] 2))"
    End
  End
End

```

```

Begin PaneConfiguration = 12
  NumPanes = 1
  Configuration = "(H (1) )"
End
Begin PaneConfiguration = 13
  NumPanes = 1
  Configuration = "(V (4))"
End
Begin PaneConfiguration = 14
  NumPanes = 1
  Configuration = "(V (2))"
End
ActivePaneConfig = 0
End
Begin DiagramPane =
  Begin Origin =
    Top = 0
    Left = 0
  End
  Begin Tables =
    Begin Table = "ZONE_KAYITLAR"
      Begin Extent =
        Top = 6
        Left = 38
        Bottom = 135
        Right = 208
      End
      DisplayFlags = 280
      TopColumn = 0
    End
  End
End
Begin SQLPane =
End
Begin DataPane =
  Begin ParameterDefaults = ""
  End
End
Begin CriteriaPane =
  Begin ColumnWidths = 11
    Column = 1440
    Alias = 900
    Table = 1170
    Output = 720
    Append = 1400
    NewValue = 1170
    SortType = 1350
    SortOrder = 1410
    GroupBy = 1350
    Filter = 1350
  End

```

```

Or = 1350
Or = 1350
Or = 1350
End
End
End
'
', @level0type=N'SCHEMA',@level0name=N'dbo',
@level1type=N'VIEW',@level1name=N'ZONE_CANLI'
GO
EXEC sys.sp_addextendedproperty @name=N'MS_DiagramPaneCount', @value=1 ,
@level0type=N'SCHEMA',@level0name=N'dbo',
@level1type=N'VIEW',@level1name=N'ZONE_CANLI'
GO
EXEC sys.sp_addextendedproperty @name=N'MS_DiagramPane1',
@value=N'[0E232FF0-B466-11cf-A24F-00AA00A3EFFF, 1.00]
Begin DesignProperties =
Begin PaneConfigurations =
Begin PaneConfiguration = 0
NumPanes = 4
Configuration = "(H (1[40] 4[20] 2[20] 3) )"
End
Begin PaneConfiguration = 1
NumPanes = 3
Configuration = "(H (1 [50] 4 [25] 3))"
End
Begin PaneConfiguration = 2
NumPanes = 3
Configuration = "(H (1 [50] 2 [25] 3))"
End
Begin PaneConfiguration = 3
NumPanes = 3
Configuration = "(H (4 [30] 2 [40] 3))"
End
Begin PaneConfiguration = 4
NumPanes = 2
Configuration = "(H (1 [56] 3))"
End
Begin PaneConfiguration = 5
NumPanes = 2
Configuration = "(H (2 [66] 3))"
End
Begin PaneConfiguration = 6
NumPanes = 2
Configuration = "(H (4 [50] 3))"
End
Begin PaneConfiguration = 7
NumPanes = 1
Configuration = "(V (3))"
End
Begin PaneConfiguration = 8

```

```

    NumPanels = 3
    Configuration = "(H (1[56] 4[18] 2) )"
End
Begin PaneConfiguration = 9
    NumPanels = 2
    Configuration = "(H (1 [75] 4))"
End
Begin PaneConfiguration = 10
    NumPanels = 2
    Configuration = "(H (1[66] 2) )"
End
Begin PaneConfiguration = 11
    NumPanels = 2
    Configuration = "(H (4 [60] 2))"
End
Begin PaneConfiguration = 12
    NumPanels = 1
    Configuration = "(H (1) )"
End
Begin PaneConfiguration = 13
    NumPanels = 1
    Configuration = "(V (4))"
End
Begin PaneConfiguration = 14
    NumPanels = 1
    Configuration = "(V (2))"
End
ActivePaneConfig = 0
End
Begin DiagramPane =
    Begin Origin =
        Top = 0
        Left = 0
    End
    Begin Tables =
        Begin Table = "ZONE_KAYITLAR"
            Begin Extent =
                Top = 6
                Left = 38
                Bottom = 136
                Right = 224
            End
            DisplayFlags = 280
            TopColumn = 0
        End
    End
End
Begin SQLPane =
End
Begin DataPane =

```

```

Begin ParameterDefaults = ""
End
End
Begin CriteriaPane =
Begin ColumnWidths = 11
Column = 1440
Alias = 900
Table = 1170
Output = 720
Append = 1400
NewValue = 1170
SortType = 1350
SortOrder = 1410
GroupBy = 1350
Filter = 1350
Or = 1350
Or = 1350
Or = 1350
End
End
End
',
@level0type=N'SCHEMA',@level0name=N'dbo',
@level1type=N'VIEW',@level1name=N'ZONE_GIRIS_CIKIS'
GO
EXEC sys.sp_addextendedproperty @name=N'MS_DiagramPaneCount', @value=1 ,
@level0type=N'SCHEMA',@level0name=N'dbo',
@level1type=N'VIEW',@level1name=N'ZONE_GIRIS_CIKIS'
GO
EXEC sys.sp_addextendedproperty @name=N'MS_DiagramPane1',
@value=N'[0E232FF0-B466-11cf-A24F-00AA00A3EFFF, 1.00]
Begin DesignProperties =
Begin PaneConfigurations =
Begin PaneConfiguration = 0
NumPanes = 4
Configuration = "(H (1[40] 4[20] 2[20] 3) )"
End
Begin PaneConfiguration = 1
NumPanes = 3
Configuration = "(H (1 [50] 4 [25] 3))"
End
Begin PaneConfiguration = 2
NumPanes = 3
Configuration = "(H (1 [50] 2 [25] 3))"
End
Begin PaneConfiguration = 3
NumPanes = 3
Configuration = "(H (4 [30] 2 [40] 3))"
End
Begin PaneConfiguration = 4
NumPanes = 2

```

```

    Configuration = "(H (1 [56] 3))"
End
Begin PaneConfiguration = 5
    NumPanels = 2
    Configuration = "(H (2 [66] 3))"
End
Begin PaneConfiguration = 6
    NumPanels = 2
    Configuration = "(H (4 [50] 3))"
End
Begin PaneConfiguration = 7
    NumPanels = 1
    Configuration = "(V (3))"
End
Begin PaneConfiguration = 8
    NumPanels = 3
    Configuration = "(H (1[56] 4[18] 2) )"
End
Begin PaneConfiguration = 9
    NumPanels = 2
    Configuration = "(H (1 [75] 4))"
End
Begin PaneConfiguration = 10
    NumPanels = 2
    Configuration = "(H (1[66] 2) )"
End
Begin PaneConfiguration = 11
    NumPanels = 2
    Configuration = "(H (4 [60] 2))"
End
Begin PaneConfiguration = 12
    NumPanels = 1
    Configuration = "(H (1) )"
End
Begin PaneConfiguration = 13
    NumPanels = 1
    Configuration = "(V (4))"
End
Begin PaneConfiguration = 14
    NumPanels = 1
    Configuration = "(V (2))"
End
ActivePaneConfig = 0
End
Begin DiagramPane =
    Begin Origin =
        Top = 0
        Left = 0
    End
    Begin Tables =

```

```
Begin Table = "ZK"  
  Begin Extent =  
    Top = 6  
    Left = 38  
    Bottom = 135  
    Right = 208  
  End  
  DisplayFlags = 280  
  TopColumn = 0  
End  
Begin Table = "ZT"  
  Begin Extent =  
    Top = 6  
    Left = 246  
    Bottom = 135  
    Right = 430  
  End  
  DisplayFlags = 280  
  TopColumn = 0  
End  
Begin Table = "P"  
  Begin Extent =  
    Top = 6  
    Left = 468  
    Bottom = 135  
    Right = 638  
  End  
  DisplayFlags = 280  
  TopColumn = 0  
End  
End  
Begin SQLPane =  
End  
Begin DataPane =  
  Begin ParameterDefaults = ""  
  End  
End  
Begin CriteriaPane =  
  Begin ColumnWidths = 11  
    Column = 1440  
    Alias = 900  
    Table = 1170  
    Output = 720  
    Append = 1400  
    NewValue = 1170  
    SortType = 1350  
    SortOrder = 1410  
    GroupBy = 1350  
    Filter = 1350
```

```

Or = 1350
Or = 1350
Or = 1350
End
End
End
'
,
@level0type=N'SCHEMA',@level0name=N'dbo',
@level1type=N'VIEW',@level1name=N'ZONE_RAPOR_VIEW'
GO
EXEC sys.sp_addextendedproperty @name=N'MS_DiagramPaneCount', @value=1 ,
@level0type=N'SCHEMA',@level0name=N'dbo',
@level1type=N'VIEW',@level1name=N'ZONE_RAPOR_VIEW'
GO
EXEC sys.sp_addextendedproperty @name=N'MS_DiagramPane1',
@value=N'[0E232FF0-B466-11cf-A24F-00AA00A3EFFF, 1.00]
Begin DesignProperties =
Begin PaneConfigurations =
Begin PaneConfiguration = 0
NumPanes = 4
Configuration = "(H (1[40] 4[20] 2[20] 3) )"
End
Begin PaneConfiguration = 1
NumPanes = 3
Configuration = "(H (1 [50] 4 [25] 3))"
End
Begin PaneConfiguration = 2
NumPanes = 3
Configuration = "(H (1 [50] 2 [25] 3))"
End
Begin PaneConfiguration = 3
NumPanes = 3
Configuration = "(H (4 [30] 2 [40] 3))"
End
Begin PaneConfiguration = 4
NumPanes = 2
Configuration = "(H (1 [56] 3))"
End
Begin PaneConfiguration = 5
NumPanes = 2
Configuration = "(H (2 [66] 3))"
End
Begin PaneConfiguration = 6
NumPanes = 2
Configuration = "(H (4 [50] 3))"
End
Begin PaneConfiguration = 7
NumPanes = 1
Configuration = "(V (3))"
End
Begin PaneConfiguration = 8

```



```

    NumPanes = 3
    Configuration = "(H (1[56] 4[18] 2) )"
End
Begin PaneConfiguration = 9
    NumPanes = 2
    Configuration = "(H (1 [75] 4))"
End
Begin PaneConfiguration = 10
    NumPanes = 2
    Configuration = "(H (1[66] 2) )"
End
Begin PaneConfiguration = 11
    NumPanes = 2
    Configuration = "(H (4 [60] 2))"
End
Begin PaneConfiguration = 12
    NumPanes = 1
    Configuration = "(H (1) )"
End
Begin PaneConfiguration = 13
    NumPanes = 1
    Configuration = "(V (4))"
End
Begin PaneConfiguration = 14
    NumPanes = 1
    Configuration = "(V (2))"
End
ActivePaneConfig = 0
End
Begin DiagramPane =
    Begin Origin =
        Top = 0
        Left = 0
    End
    Begin Tables =
        Begin Table = "ZONE_GIRIS_CIKIS"
            Begin Extent =
                Top = 6
                Left = 38
                Bottom = 119
                Right = 224
            End
            DisplayFlags = 280
            TopColumn = 0
        End
    End
End
Begin SQLPane =
End
Begin DataPane =

```

```

Begin ParameterDefaults = ""
End
End
Begin CriteriaPane =
Begin ColumnWidths = 12
Column = 1440
Alias = 900
Table = 1170
Output = 720
Append = 1400
NewValue = 1170
SortType = 1350
SortOrder = 1410
GroupBy = 1350
Filter = 1350
Or = 1350
Or = 1350
Or = 1350
End
End
End
'
', @level0type=N'SCHEMA',@level0name=N'dbo',
@level1type=N'VIEW',@level1name=N'ZONE_SURE'
GO
EXEC sys.sp_addextendedproperty @name=N'MS_DiagramPaneCount', @value=1 ,
@level0type=N'SCHEMA',@level0name=N'dbo',
@level1type=N'VIEW',@level1name=N'ZONE_SURE'
GO

```

Web Servisleri Kodları grafik.asp

```
<%
BASTAR = Request.QueryString("BASTAR")
BITTAR = Request.QueryString("BITTAR")
BASTAR1=Split(BASTAR,"-")
BITTAR1=Split(BITTAR,"-")
BASTAR2=BASTAR1(2)&"."&BASTAR1(1)&"."&BASTAR1(0)&" 00:00:00"
BITTAR2=BITTAR1(2)&"."&BITTAR1(1)&"."&BITTAR1(0)&" 23:59:59"
Dim objConn
Set objConn = Server.CreateObject("ADODB.Connection")
objConn.ConnectionString = "Provider=SQLOLEDB; Data Source=; Database=;
User ID=; Password="
objConn.Open
SORGU = "DECLARE @basla nvarchar(max) = '&BASTAR2&' DECLARE
@bitis nvarchar(max) = '&BITTAR2&' select distinct pk.ADI_SOYADI,(SELECT
SUM((DATEPART(SECOND, SURE) + 60 * DATEPART(MINUTE, SURE)) +
3600 * DATEPART(HOUR, SURE)) AS SURE FROM
dbo.ZONE_GIRIS_CIKIS left outer join ZONE_TANIM on
ZONE_GIRIS_CIKIS.CIHAZ_ID=ZONE_TANIM.ZONE_CIHAZ left outer join
PERSONEL on ZONE_GIRIS_CIKIS.ETIKET_ID=PERSONEL.ETIKET_ID
WHERE (GIRIS_ZAMAN BETWEEN @basla AND @bitis) and
(PERSONEL.ADI_SOYADI like pk.ADI_SOYADI) and (ZONE_ADI like
'ATOLYE') GROUP BY ZONE_GIRIS_CIKIS.ETIKET_ID, CIHAZ_ID,
ZONE_ADI,ADI_SOYADI ) as ZONE1, ( SELECT
SUM((DATEPART(SECOND, SURE) + 60 * DATEPART(MINUTE, SURE)) +
3600 * DATEPART(HOUR, SURE)) AS SURE FROM
dbo.ZONE_GIRIS_CIKIS left outer join ZONE_TANIM on
ZONE_GIRIS_CIKIS.CIHAZ_ID=ZONE_TANIM.ZONE_CIHAZ left outer join
PERSONEL on ZONE_GIRIS_CIKIS.ETIKET_ID=PERSONEL.ETIKET_ID
WHERE (GIRIS_ZAMAN BETWEEN @basla AND @bitis) and
(PERSONEL.ADI_SOYADI like pk.ADI_SOYADI) and (ZONE_ADI like 'OFIS')
GROUP BY ZONE_GIRIS_CIKIS.ETIKET_ID, CIHAZ_ID,
ZONE_ADI,ADI_SOYADI) as ZONE2,(SELECT
SUM((DATEPART(SECOND, SURE) + 60 * DATEPART(MINUTE, SURE)) +
3600 * DATEPART(HOUR, SURE)) AS SURE FROM
dbo.ZONE_GIRIS_CIKIS left outer join ZONE_TANIM on
ZONE_GIRIS_CIKIS.CIHAZ_ID=ZONE_TANIM.ZONE_CIHAZ left outer join
PERSONEL on ZONE_GIRIS_CIKIS.ETIKET_ID=PERSONEL.ETIKET_ID
WHERE (GIRIS_ZAMAN BETWEEN @basla AND @bitis) and
(PERSONEL.ADI_SOYADI like pk.ADI_SOYADI) and (ZONE_ADI like 'ZONE3')
GROUP BY ZONE_GIRIS_CIKIS.ETIKET_ID, CIHAZ_ID,
ZONE_ADI,ADI_SOYADI) as ZONE3 from ZONE_KAYITLAR left outer join
PERSONEL as pk on ZONE_KAYITLAR.ETIKET_ID=pk.ETIKET_ID WHERE
(GIRIS_ZAMAN BETWEEN @basla AND @bitis)"
Set DATA = objConn.Execute(SORGU)
%>
<html>
```

```

<head>
  <script src="anychart.min.js" type="text/javascript"></script>
  <script src="data-adapter.min.js" type="text/javascript"></script>
  <script src="wines.min.js" type="text/javascript"></script>
  <script>anychart.onDocumentLoad(function() {
    anychart.theme(anychart.themes.wines);
    var tableData = anychart.data.parseHtmlTable('#tableWithData');
    var chart = anychart.bar();
    chart.data(tableData);
    chart.yScale().stackMode('percent');
    var legend = chart.legend();
    legend.enabled(true);
    var tooltip = chart.tooltip();
    tooltip.displayMode('union');
    tooltip.textFormatter('{ % SeriesName }           { % Value }sn
    ({ % YPercentOfCategory }%');
    chart.container('container');
    chart.draw();
  });</script>
</head>

<body>
  <style type="text/css">
    * {
    margin: 0;
  }

#container, html, body {
  width: 100%;
  height: 100%;
}

#container {
  float: left;
  width: 100%;
}
#tableWithData {
  display: none;
}

  </style>
  <div id="container"></div>
  <table id="tableWithData" border="1" cellpadding="5">
    <caption><% Response.Write BASTAR2%> - <% Response.Write BITTAR2%>
    Tarihli Personel Performans Raporu </caption>
    <thead>
      <tr>
        <th>Personel Adı</th>
        <th>Zone-1</th>

```

```

        <th>Zone-2</th>
        <th>Zone-3</th>
    </tr>
</thead>
<tbody>
    <%
    If DATA.eof and DATA.bof Then
    Response.Write "(Kayıt Yok)<br />"
    Else
        DO WHILE Not DATA.Eof
            Response.Write "<tr>"
            Response.Write
" <td>"&DATA("ADI_SOYADI")&"</td>"
            Response.Write "<td>"&DATA("ZONE1")&"</td>"
            Response.Write "<td>"&DATA("ZONE2")&"</td>"
            Response.Write "<td>"&DATA("ZONE3")&"</td>"
            Response.Write "</tr>"
            DATA.movenext
        LOOP
    End if
    %>
</tbody>
</table>
</body>
</html>
<% objConn.Close%>

```

Web Servisleri Kodları rapor.asp

```
<form action="grafik.asp" method="get" target="my_frame">
  <i><strong>Tarih Aralığı:</strong></i><input type="date" name="BASTAR">
  <i><strong>- </strong></i><input type="date" name="BITTAR">
  <input name="gonder" type="submit" value="Hazırla" />
</form>

<iframe name="my_frame" width="100%" height="90%">
```



Web Servisleri Kodları zone.asp

```
<%
ETIKET_ID = Request.QueryString("ETIKET_ID")
CIHAZ = Request.QueryString("CIHAZ")
DURUM = Request.QueryString("DURUM")
response.write "ETIKET_ID="+ETIKET_ID+"<br>"
response.write "CIHAZ="+CIHAZ+"<br>"
response.write "DURUM="+DURUM+"<br>"

Dim objConn
Set objConn = Server.CreateObject("ADODB.Connection")
objConn.ConnectionString = "Provider=SQLOLEDB; Data Source=; Database=;
User ID=; Password="
objConn.Open
Dim SQL
if durum=1 then
SQL = "INSERT INTO [dbo].[ZONE_KAYITLAR]
([ETIKET_ID],[CIHAZ_ID],[GIRIS_ZAMAN])
VALUES
('"+ETIKET_ID+"','"+CIHAZ+"',GETDATE())"
else
SQL = "UPDATE [dbo].[ZONE_KAYITLAR] set [CIKIS_ZAMAN]=GETDATE()
where ETIKET_ID like '"+ETIKET_ID+"' and CIHAZ_ID like '"+CIHAZ+"' and
[CIKIS_ZAMAN] is null"
end if
objConn.Execute SQL
objConn.Close

%>
```