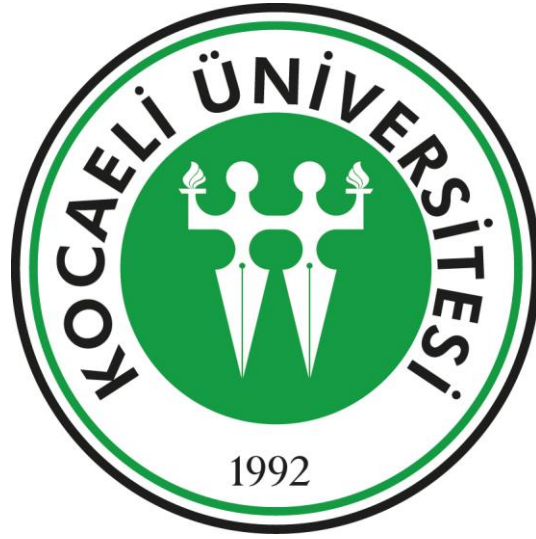


KOCAELİ ÜNİVERSİTESİ MÜHENDİSLİK FAKÜLTESİ



PRATİK ÇALIŞMA DEFTERİ

ÇALIŞMAYI YAPAN ÖĞRENCİNİN	
BÖLÜMÜ	: Bilgisayar Mühendisliği (İ.Ö.)
ADI, SOYADI	: Ersin KALAFAT
NO	: 130202117



KOCAELİ ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
Pratik Çalışma Defteri

PRATİK ÇALIŞMA DALI : Yazılım

SOYADI, ADI ve NO : Kalafat Ersin 130202117

BÖLÜM / SINIF : Bilgisayar Mühendisliği - 3

KURUMA AİT BİLGİLER

KURUMUN ADI	KISIM	ÇALIŞMA SÜRESİ		HAFTA
		Tarihinden	Tarihine	
MİRSİS Bilgi Teknolojileri	KOÜ Teknopark	11.07.2016	26.08.2016	7 Hafta

ÖĞRENCİNİN DURUMU

TAVIR VE HAREKETLER,

ÇALIŞKANLIK,

YETENEK,

1/ KONTROL EDEN AMİR, ÖĞRENCİNİN PRATİK SIRASINDAKİ DURUMUNU BİLDİRİR.

2/ YUKARIDA YAZILI TAVIR ve HAREKETLER, ÇALIŞKANLIK, YETENEK DURUMLARININ TASNİFİ

A/ PEKİYİ

B/ İYİ

C/ ORTA

D/ GEÇER

F/ KÖTÜ

KOCAELİ ÜNİVERSİTESİ

STAJ ESASLARI

1. Her bölümün Staj Komisyonu, en az bir öğretim üyesi olmak üzere üç öğretim elemanından oluşur. Komisyon üyeleri, Bölüm Başkanı tarafından iki yıl için seçilerek Dekanlığa bildirilir.
2. Bölümlere verilen staj kontenjanları, staj komisyonları tarafından öğrencilere başarı ve mezuniyet durumları dikkate alınarak dağıtılır.
3. Kendilerine staj yeri tahsis edilen öğrenci mazeretsiz olarak staja gitmezse, daha sonraki yıllarda ancak artan kontenjanlardan yararlandırılabilir.
4. Staj yerlerini kendileri temin eden öğrenciler, iş yerlerinden staja kabul edildiklerine ilişkin bir yazı getirerek Bölüm Staj Komisyonuna onaylatırlar. Staj yapılacak iş yerinin, staj yapmaya uygun olup olmadığı, her bölümün özel hükümlerinde belirlenir.
5. Öğrenciler staja, staj defteri ve staj sicil fişi ile Öğrenci İşleri Bürosu tarafından sevk edilirler.
6. Öğrenciler çalıştıkları iş yerlerinin kanun, tüzük, yönetmelik ve kurallarına uymak zorundadırlar.
7. Öğrenciler stajlarını sırasına uygun olarak yaparlar
8. Prensip olarak her staj sürekli olup, ait olduğu dönemde ve bir defada yapılır. Ancak Staj Komisyonunca uygun görülen geçerli bir mazeretten dolayı yarıda kalan stajlar aynı veya bir sonraki dönemde tamamlattırılır.
9. Staj yerlerinde yapılacak işlemlerle ilgili ayrıntılı bilgileri içeren yönergeler, öğrencinin staj yapacağı kuruma öğrenciler tarafından iletilmek zorundadır. (Staj fişinin arkasında kurumca yapılacak işlemler belirtilecektir.)
10. Stajlar esnasında yapılan tüm işlemlerin teknik detayı gerekli resim ve hesaplar temiz bir şekilde staj defterine günü gününe kaydedilir.
11. Öğrenci çalışma konusunu tanıtmak üzere fotoğraf, fotokopi, ozalit, rapor, proje vs. gibi dokümanları staj defterine ekleyebilir.
12. Staj defterinde belirtilen kısımlar eksiksiz doldurulup, ilgili sorumlulara imzalatılır. Defter iş yeri yetkilisi tarafından onaylanır.
13. Staj sicil fişi, iş yeri yetkilileri tarafından doldurulup tasdik edilerek, ilgili Bölüm Başkanlığına posta ile kapalı ve mühürlü zarf ile gönderilir.
14. Bölüm staj komisyonu gerek gördüğü takdirde öğrencileri staj yerinde denetler.
15. Öğrenci staj defterini güz yarıyılıının ilk iki haftası içerisinde, öğretim yılı içinde yapılan stajlarda ise stajın bitim tarihinden itibaren iki hafta içinde Bölüm Staj Komisyonu Başkanlığına teslim etmek zorundadır.
16. Değerlendirme sırasında öğrenciler stajları ile ilgili olarak, Staj Komisyonunca görüşmeye tabi tutulurlar, staj defterleri ve ilgili raporlar değerlendirilerek öğrencilerin başarılı veya başarısız olduklarına karar verilir.
17. Kabul veya reddedilen stajlar Bölüm Başkanlığınca ilan edilir.
18. Staj sicil fişi, staj defteri ve Staj Komisyonunca doldurulan staj kabul formu öğrencinin dosyasında muhafaza edilir.

11.07.2016 TARİHİNDEN, 15.07.2016 TARİHİNE KADAR 1 HAFTALIK ÇALIŞMA			
GÜN	YAPILAN İŞLER	SAYFA NO:	ÇALIŞILAN SAAT:
PAZARTESİ	Tanışma, İnovasyon Yönetimi Eğitimi, Türkiye’de Teknoloji Yönetimi, Fintech	8	10
SALI	BlockChain Algoritması, Bitcoin	9	10
ÇARŞAMBA	Çevik Yöntemler, Proje Belirleme	10	10
PERŞEMBE	Proje teknolojileri, Android Programlama, Tasarım, Teknik hata giderme	11	10
CUMA	Proje Güncelleme, Bireysel bölümü, Raporlama	12	10
CUMARTESİ			
KISIM:		TOPLAM SAAT: 50	
ÇALIŞTIĞI İŞYERİ VE KISIM: Mirsis Bilgi Teknolojileri, KOÜ Teknopark Ofisi			
KONTROL EDENİN ÜNVANI:			
SOYADI, ADI VE İMZASI:			

18.07.2016 TARİHİNDEN, 22.07.2016 TARİHİNE KADAR 1 HAFTALIK ÇALIŞMA			
GÜN	YAPILAN İŞLER	SAYFA NO:	ÇALIŞILAN SAAT:
PAZARTESİ	Akbank API’leri Sunumu, ORM, Cloud Servisler, Temiz Kod Sunumu, Zaman Yönetimi	13	10
SALI	Bulut Bilişim, Proje İyileştirme	14	10
ÇARŞAMBA	Digitalocean, Web Servisler (SOAP, RESTful)	15	10
PERŞEMBE	Microsoft Azure, Veritabanı, Sunucu, Visual Studio 2015	16	10
CUMA	Microsoft Azure, Web Servis	17	10
CUMARTESİ			
KISIM:		TOPLAM SAAT: 50	
ÇALIŞTIĞI İŞYERİ VE KISIM: Mirsis Bilgi Teknolojileri, KOÜ Teknopark Ofisi			
KONTROL EDENİN ÜNVANI:			
SOYADI, ADI VE İMZASI:			

25.07.2016 TARİHİNDEN, 29.07.2016 TARİHİNE KADAR 1 HAFTALIK ÇALIŞMA			
GÜN	YAPILAN İŞLER	SAYFA NO:	ÇALIŞILAN SAAT:
PAZARTESİ	Haftalık Proje Sunumu, Design Patterns, Çok Katmanlı Mimari	18	10
SALI	Proje düzenlemeleri, Personel Login, Patent İşlemleri Sunumu	19	10
ÇARŞAMBA	İnsan Kaynakları Semineri	20	10
PERŞEMBE	Yönetici Login Web Servis ve Java	21	10
CUMA	Personel Ekle, Web Servis ve Java	22	10
CUMARTESİ			
KISIM:		TOPLAM SAAT: 50	
ÇALIŞTIĞI İŞYERİ VE KISIM: Mirsis Bilgi Teknolojileri, KOÜ Teknopark Ofisi			
KONTROL EDENİN ÜNVANI:			
SOYADI, ADI VE İMZASI:			

01.08.2016 TARİHİNDEN, 05.08.2016 TARİHİNE KADAR 1 HAFTALIK ÇALIŞMA			
GÜN	YAPILAN İŞLER	SAYFA NO:	ÇALIŞILAN SAAT:
PAZARTESİ	Personelleri Liste Şeklinde Görüntüleme, ListView	23	10
SALI	Personel İçerik Kısmı, ListView tıklandığında gidilecek sayfa	24	10
ÇARŞAMBA	Proje Sunumları, Başarının Sırrı Semineri	25	10
PERŞEMBE	Personel İçerik Sayfası Düzenleme, Personeler Para Gönderme	26	10
CUMA	Personel İçerik Sayfası Düzenleme, Personel Silme	27	10
CUMARTESİ			
KISIM:		TOPLAM SAAT: 50	
ÇALIŞTIĞI İŞYERİ VE KISIM: Mirsis Bilgi Teknolojileri, KOÜ Teknopark Ofisi			
KONTROL EDENİN ÜNVANI:			
SOYADI, ADI VE İMZASI:			

08.08.2016 TARİHİNDEN, 12.08.2016 TARİHİNE KADAR 1 HAFTALIK ÇALIŞMA			
GÜN	YAPILAN İŞLER	SAYFA NO:	ÇALIŞILAN SAAT:
PAZARTESİ	Genel Toplantı, Modern Bankacılık Semineri	28	10
SALI	Personel İçerik Sayfası Düzenleme, Personel Harcama Durumu	29	10
ÇARŞAMBA	Personel Harcama Sayfası	30	10
PERŞEMBE	Personel Durum Sayfası, Finans Durumu	31	10
CUMA	Proje Sunumları, İş Ahlakı ve Etik, Sistem Yöneticiliği, B.M. ve İş Hayatı Sunumu	32	10
CUMARTESİ			
KISIM:		TOPLAM SAAT: 50	
ÇALIŞTIĞI İŞYERİ VE KISIM: Mirsis Bilgi Teknolojileri, KOÜ Teknopark Ofisi			
KONTROL EDENİN ÜNVANI:			
SOYADI, ADI VE İMZASI:			

15.08.2016 TARİHİNDEN, 19.08.2016 TARİHİNE KADAR 1 HAFTALIK ÇALIŞMA			
GÜN	YAPILAN İŞLER	SAYFA NO:	ÇALIŞILAN SAAT:
PAZARTESİ	Personel Durum Sayfası, Harcama Durumu	33	10
SALI	Personel Harcama Geçmişi	34	10
ÇARŞAMBA	ProgressBar ve Pasta Grafiği	35	10
PERŞEMBE	Proje ER Şeması Oluşturma ve Kodların Ayrıntılı Düzenlenmesi	36	10
CUMA	XML Dizayn : İntro, Yönetici Login, Personel Login	37	10
CUMARTESİ			
KISIM:		TOPLAM SAAT: 50	
ÇALIŞTIĞI İŞYERİ VE KISIM: Mirsis Bilgi Teknolojileri, KOÜ Teknopark Ofisi			
KONTROL EDENİN ÜNVANI:			
SOYADI, ADI VE İMZASI:			

22.08.2016 TARİHİNDEN, 26.08.2016 TARİHİNE KADAR 1 HAFTALIK ÇALIŞMA

GÜN	YAPILAN İŞLER	SAYFA NO:	ÇALIŞILAN SAAT:
PAZARTESİ	XML Dizayn : Yönetici Üyelik, Personel Ekleme	38	10
SALI	XML Dizayn : Personel İçerik	39	10
ÇARŞAMBA	XML Dizayn : Personel Listesi, Harcama Sayfası, Geçmiş	40	10
PERŞEMBE	XML Dizayn : Personel Finans ve Harcama Durumları Sayfası	41	10
CUMA	Sunum Oluşturma	42, 43	10
CUMARTESİ			
KISIM:		TOPLAM SAAT: 50	
ÇALIŞTIĞI İŞYERİ VE KISIM: Mirsis Bilgi Teknolojileri, KOÜ Teknopark Ofisi			
KONTROL EDENİN ÜNVANI:			
SOYADI, ADI VE İMZASI:			

GÜN	YAPILAN İŞLER	SAYFA NO:	ÇALIŞILAN SAAT:
PAZARTESİ			
SALI			
ÇARŞAMBA			
PERŞEMBE			
CUMA			
CUMARTESİ			
KISIM:		TOPLAM SAAT:	
ÇALIŞTIĞI İŞYERİ VE KISIM:			
KONTROL EDENİN ÜNVANI:			
SOYADI, ADI VE İMZASI:			

❖ **Şirket tanıma ve ekip arkadaşlarıyla tanışma gerçekleştirildi.**

❖ **Giriş olarak İnovasyon Yönetimi eğitimi verildi.**

- İnovasyon Nedir?
- İnovasyon, yeni veya iyileştirilmiş ürün, hizmet veya üretim yöntemi geliştirmek ve bunu ticari gelir elde edecek hale getirmek için yürütülen tüm süreçleri kapsar. Yeni veya iyileştirilmiş ürün, hizmet veya üretim yöntemi geliştirme, yeni düşüncelerden doğar. İnovasyon, sürekliliği olan bir faaliyettir.

❖ **Türkiye’de teknoloji yönetimi**

- Teknoloji geliştirme süreçleri
- Teknolojinin ticarileştirilmesi ve inovasyon
- Proje gerçekleştirme süreçleri
- Teknoloji yönetimi yeteneği

❖ **Fintech Sektörü**

- Teknoloji ve finans dünyasının bir araya gelmesiyle ortaya çıkan finansal teknolojiler, kısaca Fin-Tech son yıllarda hızla gelişmekte olan bir sektördür.
- Dünya’da Fintech sektöründeki firmaların sayısı gittikçe artmaktadır. Türkiye’de bu sayı oldukça az olduğundan bu alanda Türkiye’de büyük bir Pazar bekliyor.
- Finansal Teknoloji Yatırımları
- Banka Erişimi Olmayan Ülkeler
- İlk Fintech Toplantısı Ocak 2016 yılında gerçekleştirilmiştir

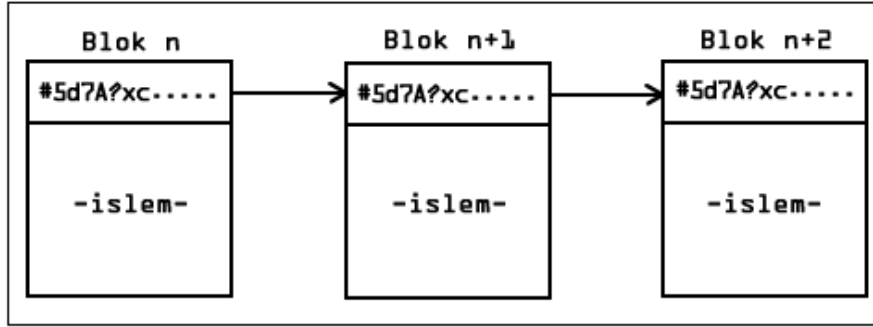


KONTROL SONUCU:

YAPILAN İŞ: BlockChain Algoritması, Bitcoin

❖ BlockChain Algoritması

- Kriptolojik bir şekilde varlıkların şifrelendiği, tek bir merkez veritabanına bağlı olmayan, aslında herkesin erişebildiği sabit olmayan bir veritabanı yapısıdır. Linklist ve Hash kavramlarının birleşiminden doğan bir yapıdır.



Çizim : Ersin Kalafat

❖ Bitcoin

- Paranın geleceği Bitcoin mi?
- Herhangi bir merkez bankası, resmi kuruluş, vs. ile ilişkisi olmayan elektronik bir para birimidir . Virgülden sonra 100 milyonuncu basamağa kadar birimlere ayrılabilir.
- Bitcoin, 3 Ocak 2009’da hayata geçen, BlockChain algoritmasının bir türevidir.
- Bitcoin’de gerçekleşen ödemeler diğer noktalara anında ulaşır, hangi adresten hangi adrese ödeme yapıldığı kayıtlara geçer. Böylece toplanan kayıtlar blok adı verilen yapıların içinde yer alır. Her bir blok üzerinde yüksek işlem gerektiren bir hash algoritması uygulanarak, belli bir sıfır sayısı ile başlayan ifadenin bulunması istenmektedir.
- Para yaratma işlemine madencilik (mining) denilmektedir. Madencilik, hesaplama yetkisini ve gücünü kullanarak matematiksel işlemleri gerçekleştirme işleminin genel adıdır.
- Bu sistemde paranın geçmişini, kaynağını takip edebiliriz. Bu teknolojinin ilerde tapu, noterlik, oylama sistemleri gibi ortamlarda kullanılabilmesi hayal ediliyor.



KONTROL SONUCU:

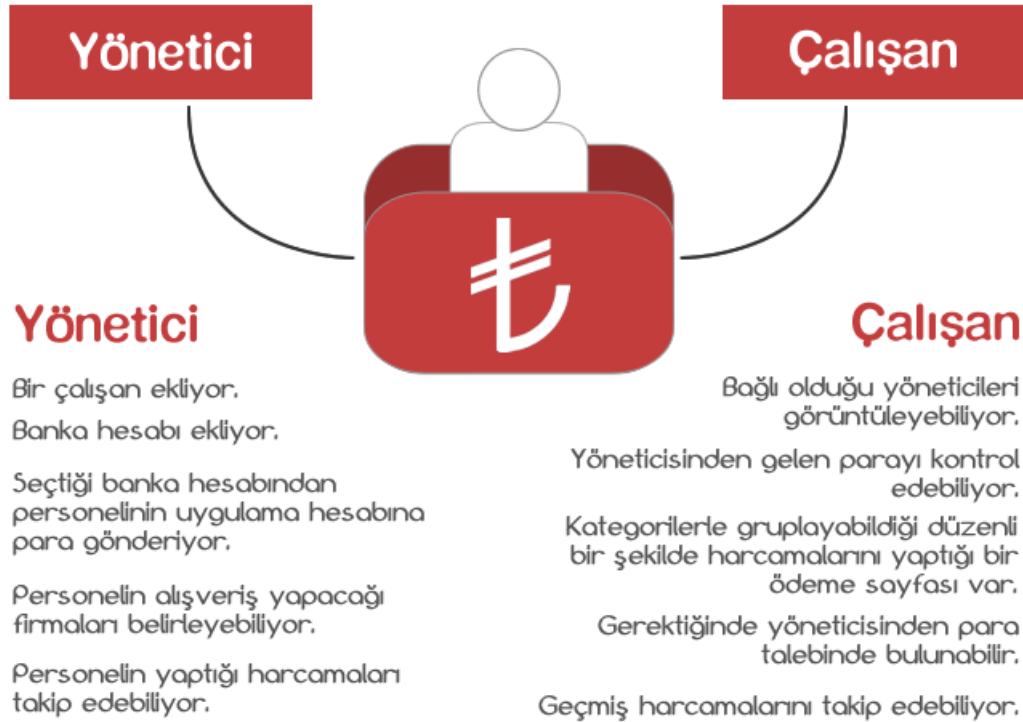
YAPILAN İŞ: Çevik Yöntemler, Proje Belirleme

❖ Çevik Yöntemler

- Yazılım dünyasında başı çeken bazı yazılımcı büyüklerinin toplanıp yazılım geliştirme süreçlerini belli başlı standartlara bağlamasıyla oluşmuştur. Dünya üzerinde kabul gören bu çevik yöntemler en hızlı, en verimli ve en güvenli proje geliştirme yaklaşımıdır.
- Çevik yöntemlerden en yaygın olarak kullanılan SCRUM metodolojisiyle ilgili bilgi toplandı. Bu metodu kullanarak bir proje geliştirmemiz istendi.

❖ Proje Başlama

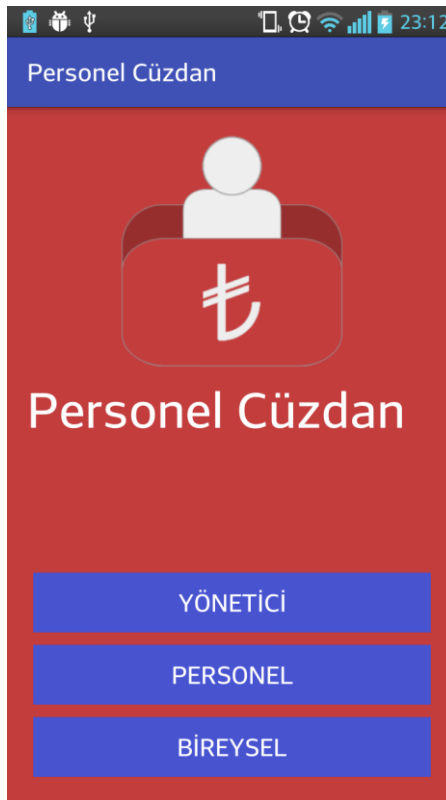
- Staja başlamadan önce bizden proje önerileri istendi.
- Projemiz kendi fikrimiz olan ve firma tarafından da onaylanarak istenen fintech üzerine bir Android mobil uygulama.
- Proje özetle mobil cüzdan mantığıyla çalışacak bir çalışan harcama bütçesi yöneticisi.
- Ekte proje üzerine fikirler atıldı, eklemeler, çıkarmalar yapıldı. Ardından projede izlenecek adımlar ve hedef sabitleştirildi.
- Logo tasarımı yapıldı.



KONTROL SONUCU:

❖ **Proje**

- Proje için kullanılacak teknolojiler belirlendi. Uygulamamızın geliştirme ortamı için Android Studio kullanmaya karar verdik. Android Studio, SDK, JDK ve emülatör eklentisi olarak daha kullanışlı ve hızlı olduğu için Genymotion kurulumunu gerçekleştirdik.
- Projemizi SCRUM metoduna göre geliştirdiğimizden ilk hafta bitiminde elimizde basit de olsa çalışan bir şeyler olması gerekiyor. Bu sebepten Yönetici ve Personel kısımlarında yapacaklarımızı geçici olarak Bireysel kısmında yapmaya karar verdik. Yani başlangıç olarak para aktarma, para harcama ve harcama raporu görüntüleme işlerini bu sayfada yapacağız.
- Giriş olarak uygulamamızın giriş sayfasını tasarlayıp Android'de kodladık. Giriş sayfasında bir logo, proje ismi, ve Yönetici, Personel, Bireysel isimli 3 adet buton olmasını kararlaştırdık. Uygulamamızda kullanacağımız renkleri belirledik. Uygulamanın giriş sayfasını buna göre oluşturduk.
- Buraya kadar JAVA, XML ve Fireworks kullandık.
- Geliştirme ortamıyla ve emülatörle ilgili karşılaştığımız bazı teknik hataları giderdik.



KONTROL SONUCU:

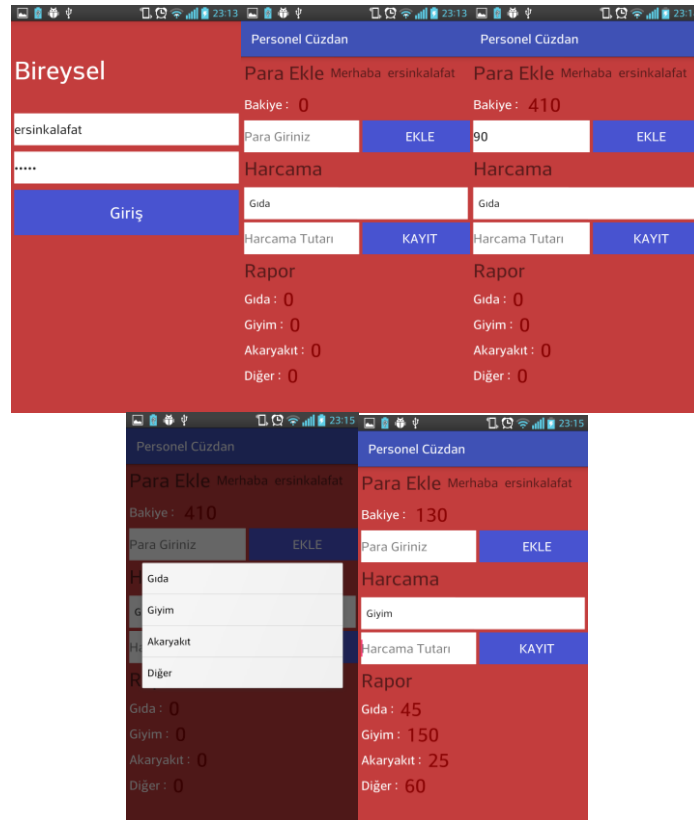
YAPILAN İŞ: Proje Güncelleme, Bireysel bölümü, Raporlama

❖ Proje

- Projenin mevcut durumunda güncellemeler yapıldı. Bireysel giriş sayfası tasarlandı. Kullanıcının Bireysel olarak sisteme giriş yaptıktan sonraki kendisine bakiye ekleyebileceği seçtiği kategoriye göre (“Gıda”, “Giyim”, “Akaryakıt”, “Diğer”) harcama yapabileceği bir arayüz oluşturuldu. Oluşturulan arayüz henüz web servis, sunucu, veri tabanı gibi işlemler yapılmadığından şu an sadece arka plan hesaplamalarıyla kullanıcı deneyimi sunuyor.

❖ Raporlama

- SCRUM metoduna göre 2 haftada bir rapor oluşturulmakta. Ancak bizim programımız kısa olduğu için haftalık rapor istendi.
- Yazılım Müdürümüz Ersin Beye haftalık raporumuzu yazılı bir şekilde e-mail yoluyla ilettik.
- Rapor içeriğine bu hafta neler yaptığımızı, hangi sorunlarla karşılaştığımızı ve gelecek hafta neler yapacağımızı yazdık.



KONTROL SONUCU:

❖ **Akbank API Portal**

- Staj yapmış olduğum firma finansal teknolojilerle ilgileniyor ve Akbank ile bazı çalışmaları var. Akbank, Dünyada API'lerini açan ilk banka oldu. Stajdaki 6 proje grubundan 4 grup Akbank API'lerini kullanarak uygulama geliştiriyor. Bu nedenle Akbank API'lerinin nasıl kullanacağıyla ilgili eğitim niteliğinde bir sunum gerçekleştirildi.

❖ **ORM (Object Relational Mapping)**

- İlişkisel veri tabanı ile nesneye yönelik programlama arasında bir tür köprü özelliği gören ve ilişkisel veri tabanındaki bilgilerimizi yönetmek için nesne modellerimizi kullandığımız bir tekniktir.

❖ **Rapor Değerlendirme**

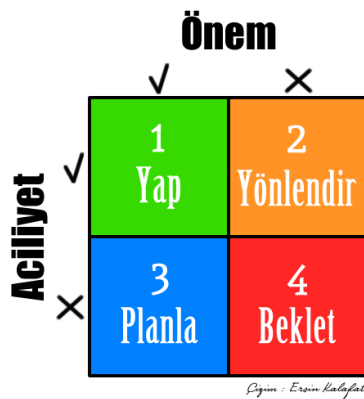
- Yazılım Müdürümüz Ersin Bey, projelerin haftalık raporları üzerinden değerlendirmeler ve önerilerde bulundu.

❖ **Temiz Kod**

- Ersin Bey "Temiz Kod" üzerine bir sunum gerçekleştirdi.
- Temiz kod nasıl olmalı?
- Kod yazarken nelere dikkat edilmeli?
- Yorum satırı gerekli/faydalı mı?
- Kodun temizliğini ölçen PMD programları

❖ **Zaman Yönetimi**

- Zamanı verimli ve etkili kullanmayı amaçlamalıyız.



- Bazen şirket içerisinde atmosferi bozmamak ve işlem geçmişini saklayabilmek için işle ilgili işlemler için çalışma arkadaşıyla konuşarak anlaşmak yerine mail ile haberleşmek tercih edilebilir.

- İş hayatında bazen geribildirim alacağımız zamanlar olacak. Bu geribildirimler aldığımız üst makamın fikirlerine saygı duymalı, kişiyi dinlemeli ve gerekirse değişime açık olmalıyız.

KONTROL SONUCU:

YAPILAN İŞ: Bulut Bilişim, Proje İyileştirme

❖ **Bulut Bilişim**

- Uygulamalarımızda veri tabanı olarak bulut sistemi kullanmamız ve verilerimizi kaydedip çekme gibi işlemleri web servisle bulut veritabanına bağlanarak yapmamız istendi. Bu konuda çeşitli araştırmalar yaptım.
- Araştırma sonucunda aşağıdaki verileri elde ettim.

Bulut Platformu	Uyumluluk	Ücret
Digitalocean	Linux	Ucuz (2GB Memory, 40GB SSD, 3TB Transfer = 20 Dolar)
Microsoft Azure	Windows	30 Gün Deneme Sürümü (Sonrası Ücretli veya Dreamspark ile kısıtlı olarak ücretsiz)
Turkcell Datagrid	Windows	Aktif Değil
Amazon Cloud Service	Windows	30 Gün Deneme Sürümü (Sonrası Ücretli)
Google Cloud	Windows	60 Gün Deneme Sürümü (Sonrası Ücretli)
Mongo DB ve ElephantDB	Windows	Ücretsiz
Softlayer (IBM Cloud)	Windows	30 Gün Deneme Sürümü (Sonrası Ücretli)

❖ **Proje**

- Uygulamamızda henüz bulutumuzu seçmesek de SCRUM metodu uyguladığımızdan dolayı işe yarar, müşteriye somut bir şeyler verecek şekilde bazı değişiklikler ve denemeler yapmaya başladık.
- Giriş, Para Ekle, Para Harca, Rapor kısımlarını düzenledik ve tasarım iyileştirmeleri yaptık.



KONTROL SONUCU:

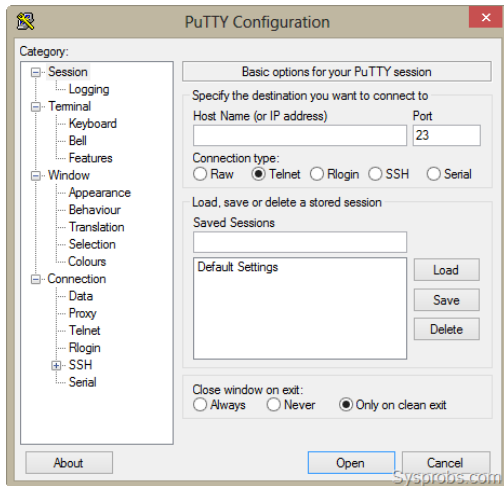
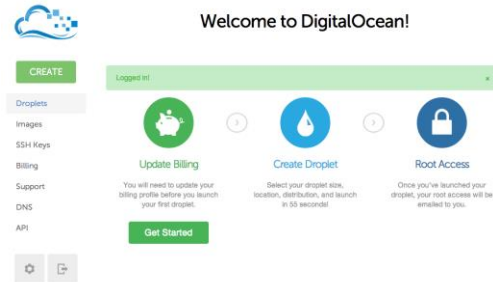
YAPILAN İŞ: Digitalocean, Web Servisler (SOAP, RESTful)

❖ DigitalOcean

- Cloud araştırmalarına devam ettik.
- Kullanışlılığı sebebiyle DigitalOcean bulut platformunu denedik. Masraflar için “GitHub Öğenci Paketi” fırsatından yararlandık. Paket kapsamında DigitalOcean’da 50\$’lık bir krediye sahip olduk.
- DigitalOcean’da sanal sunuculara Droplet deniyor. GitHub yardımımızdaki 50\$’lık krediyi Droplet kurulumunda kullandık.
- DigitalOcean Linux işletim sistemiyle çalıştığından veritabanı işlemleri Terminal üzerinde gerçekleştiriliyor. Bunun için Putty programını kurup kullandık. Cloud veritabanımıza bağlanmak için yollar denedik. Putty üzerinden veritabanımıza bir tablo oluşturduk ancak tablodaki verileri uygulamamıza çekemedik.

❖ SOAP - RESTful

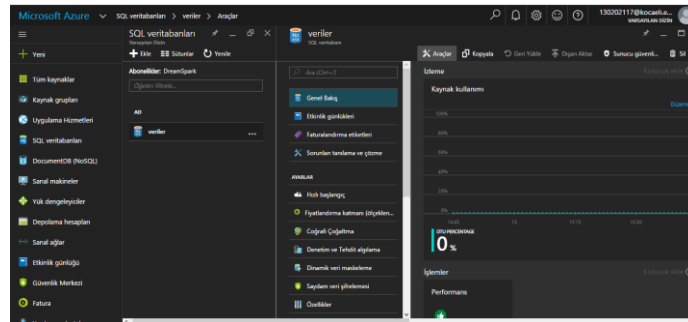
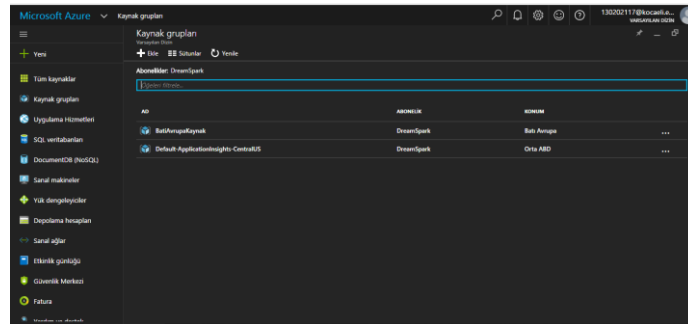
- Bunların yanı sıra Android web servis işlemleriyle ilgili araştırmalar gerçekleştirdik. SOAP ve RESTful web servislerini araştırdık. PHP ile SOAP web servisi oluşturmayla ilgili birtakım araştırmalar yaptık.



KONTROL SONUCU:

❖ Microsoft Azure

- DigitalOcean'da uğraşmaya devam ettik ancak web servis oluşturma konusunda başarısız olduk. DigitalOcean'dan vazgeçtik ve Microsoft Azure'a geçmeye karar verdik. Dreamspark.edu.tr uzantılı öğrenci hesabımızdan bir Microsoft Azure bulutu edindik.
- **Microsoft Azure adımları;**
 1. Öncelikle Microsoft Azure web sitesinden öğrenci hesabımızı tanıttık ve mail adresinden onayladık.
 2. Tarayıcımızdan Azure portalına giriş yaptık.
 3. Sol taraftaki menüden Uygulama Hizmetleri > Ekle > Web Uygulaması diyerek yeni bir web uygulaması oluşturduk.
 4. Sol taraftaki menüden Kaynak Grupları > Ekle > Kaynak Grubu diyerek yeni bir kaynak grubu oluşturduk.
 5. Sol taraftaki menüden SQL Veritabanları > Ekle > SQL Veritabanı diyerek, oluşturduğumuz Kaynak Grubunu seçerek ve yeni bir sunucu belirleyip o sunucuya bağlanarak bir SQL Veritabanı oluşturduk. Oluşturduğumuz veritabanı için kullanıcı adı ve şifre belirledik.
 6. Daha sonra bu oluşturduğumuz veritabanına bağlanmak için SQL Veritabanları > "veritabanımız" > Araçlar > Visual Studio'da Aç adımlarını uyguladık.
 7. Veritabanına Visual Studio'dan tablo ekledik.



KONTROL SONUCU:

YAPILAN İŞ: Microsoft Azure, Web Servis

- **Microsoft Azure**
- Azure platformunda oluşturduğumuz veritabanını kullanmaya koyulduk.
- Microsoft Azure'da veritabanına bağlanarak web servis oluşturmak için için neler gerektiğini araştırdık.
- Veritabanını Visual Studio'da açıp içinde tablo oluşturduktan sonra Visual Studio > New Project > ASP.NET Web Application diyerek C# dilinde bir web uygulaması oluşturduk.
- Web uygulamamızı Azure'daki veritabanımıza bağlamak için birtakım işlemler yaptık.
- [WebService(Namespace = "http://mobilcuzdan.azurewebsites.net/")] satırını yazarak adresimizi tanımladık.
- Yönetici Üyelik işlemimizi yapacak bir web servis yazdık. Web servisimizi SqlConnection sınıfını kullandık.

```
sqlBaglantisi = new SqlConnection("Data
Source=mobilcuzdan.database.windows.net;Initial Catalog=veriler;Integrated
Security=False;User ID=ersnklft;Password=*****;Connect
Timeout=60;Encrypt=False;TrustServerCertificate=True;ApplicationIntent=Read
Write;MultiSubnetFailover=False");
```
- Son olarak da aşağıdaki ekran görüntüsünü aldığım kodları yazdık ve üyelik işlemi için C# kısmını tamamladık.

```
sqlKomutu = new SqlCommand("SELECT kullanic_iadi FROM uyeler", sqlBaglantisi);
SqlDataReader veriOkuyucu = sqlKomutu.ExecuteReader();
if (veriOkuyucu != null)
{
    while (veriOkuyucu.Read())
    {
        vtKullanic_iadi = veriOkuyucu["kullanic_iadi"].ToString();
    }
}
sqlBaglantisi.Close();
if (vtKullanic_iadi.Equals(kullanic_iadi))
{
    return "Bu kullanıcı adı kullanılıyor!";
}
else
{
    sqlKomutu2 = new SqlCommand("INSERT INTO uyeler (isim, soyisim, kullanic_iadi, sifre) VALUES ('" + isim + "', '" + soyisim + "', '" + kullanic_iadi + "', '" + sifre + "')", sqlBaglantisi);
    sqlBaglantisi.Open();
    sqlKomutu2.ExecuteNonQuery();
    sqlBaglantisi.Close();
    return "Üye Kaydı Başarılı!";
}
```

- Web tarayıcıdan servisimizin adresine girdiğimizdeki görüntü.

servisCuzdan

The following operations are supported. For a formal definition, please review the [Service Description](#).

- [yoneticiekleWs](#)

- Yazılım Müdürümüz Ersin Beye haftalık raporumuzu yazılı bir şekilde e-mail yoluyla ilettik.
- Rapor içeriğine bu hafta neler yaptığımızı, hangi sorunlarla karşılaştığımızı ve gelecek hafta neler yapacağımızı yazdık.

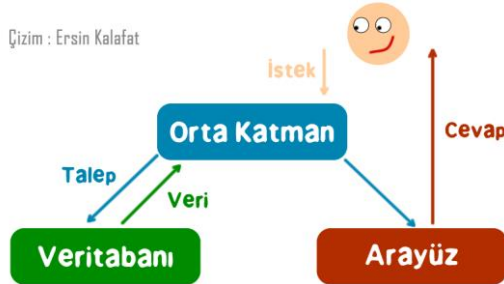
KONTROL SONUCU:

❖ **Haftalık Proje Sunumu**

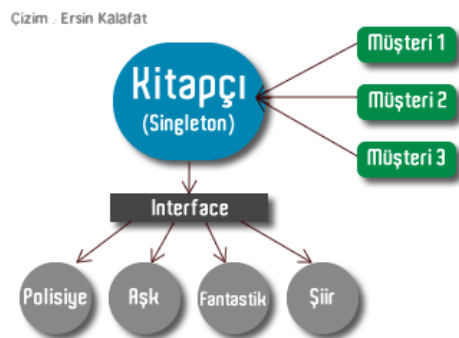
- 2 hafta sonunda projede geldiğimiz noktayı, karşılaştığımız sorunları ve hedeflerimizi anlattığımız teknik bir sunum yaptık.

❖ **Design Patterns (MVC ve Singleton)**

- Ersin Beyin Design Pattern'lerle ilgili bir sunumu oldu. Pattern, bir sorun için üretilmiş tekrar tekrar kullanılabilir bir çözümdür. Design Pattern'ler ise yazılım projelerinde sıklıkla karşılaşılan benzer sorunları çözmek için yazılımcıların işini kolaylaştıran bir yapıdır. Design Pattern'ler kod optimizasyonunu en iyi şekilde yapabilmemizi sağlar.
- MVC : Model View Controller. Günümüzde en çok kullanılan Design Pattern'lerden biridir. Bu modelde bir proje "Model", "View", "Controller" olarak üç parçaya ayrılır ve her parça ayrı ayrı geliştirilir. Bu da geliştirici açısından kolaylık sağlamaktadır.



- Singleton : Bu pattern bir programın yaşam süresince belirli bir nesneden sadece bir örneğinin olmasını garantiler. Örneğin bir veritabanı uygulamasında her gerektiğinde yeni bir bağlantı nesnesi yaratmaktansa var olan bağlantı nesnesi kullanılarak sistem kaynaklarının daha efektif bir şekilde harcanması sağlanır.



```
{ Private conn = NULL;
  getConnection(url) {
    if (conn==NULL) {
      conn = new Connection(url);
    }
    Return conn;
  }
}
```

❖ **Sıralama Algoritmaları**

- Ayrıca sunumda sıralama algoritmalarıyla ilgili de bazı çalışmalar yapıldı. Tahtada stajyerlerin de katılımıyla Selection Sort ve Bubble Sort arasındaki farkları irdeledik.

KONTROL SONUCU:

YAPILAN İŞ: Proje düzenlemeleri, Personel Login, Patent İşlemleri Sunumu

❖ Proje

- Projenin mevcut durumu incelendi.
- Kodlar temize çekildi.
- Bazı uyarlamalar yapıldı.
- Test amaçlı olarak yapılan önceki işlemleri sunucu tabanlı olarak yeniden düzenlemeye başladık.
- Kullanıcı bilgilerini (Id, Ad, Soyad, Kullanıcı Adı, Şifre, Bakiye) veritabanından web servisle çekmeye çalıştık.
- Personel için bir login işlemi gerçekleştirdik.

```

[WebMethod]
public int personelGiris(String kullaniciAdi, String sifre)
{
    String vtKullaniciAdi = null, vtSifre = null;

    con = new SqlConnection("Data Source=mobilcurdan.database.windows.net;Initial Catalog=mobilcurdan;Integrated Security=False;User
    con.Open();
    cmd = new SqlCommand("SELECT * FROM Personel WHERE kullanici_adi='" + kullaniciAdi + "' AND sifre = '" + sifre + "'", con);

    SqlDataReader reader = cmd.ExecuteReader();

    if (reader != null)
    {
        while (reader.Read())
        {
            vtKullaniciAdi = reader["kullanici_adi"].ToString().Trim();
            vtSifre = reader["sifre"].ToString().Trim();
        }
    }

    if (kullaniciAdi.Equals(vtKullaniciAdi) && sifre.Equals(vtSifre))
    {
        return 1;
    }
    else
    {
        return 0;
    }
}

```

```

public int personelGiris(String kullaniciAdi, String sifre) {
    int sonuc = 0;

    SoapObject request = new SoapObject(NAMESPACE, METROD_NAMES);
    request.addProperty("kullaniciAdi", kullaniciAdi);
    request.addProperty("sifre", sifre);

    SoapSerializationEnvelope envelope = new SoapSerializationEnvelope(SoapEnvelope.V1);
    envelope.dotNet = true;
    envelope.setOutputSoapObject(request);

    HttpTransportSE androidHttpTransportSE = new HttpTransportSE(URL);
    androidHttpTransportSE.debug = true;

    try {
        androidHttpTransportSE.call(SOAP_ACTION3, envelope);

        SoapPrimitive response = (SoapPrimitive) envelope.getResponse();
        sonuc = Integer.parseInt(response.toString());
    } catch (Exception e) {
        e.printStackTrace();
    }

    return sonuc;
}

```

```

public class mAsyncTask extends AsyncTask<Void, Void, Void> {
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressBar.setVisibility(View.VISIBLE);
    }

    @Override
    protected void doInBackground(Void... voids) {
        PersonelNS object = new PersonelNS();
        sonuc = object.personelGiris(kullaniciAdi, sifre);
        return null;
    }

    @Override
    protected void onPostExecute(Void aVoid) {
        super.onPostExecute(aVoid);
        progressBar.setVisibility(View.GONE);

        if (sonuc == 1) {
            Intent intent = new Intent(LoginPersonel.this, Personel.class);
            intent.putExtra("KADI", kullaniciAdi);
            intent.putExtra("SIFRE", sifre);
            startActivity(intent);
        }
        else if (sonuc == 0) {
            Toast.makeText(LoginPersonel.this, "Yanlış kullanıcı adı veya parola!", Toast.LENGTH_SHORT).show();
            edtPersonelKullaniciAdi.getText().clear();
            edtPersonelSifre.getText().clear();
        }
        else {
            Toast.makeText(LoginPersonel.this, "Bir hata oluştu tekrar deneyiniz.", Toast.LENGTH_SHORT).show();
            edtPersonelKullaniciAdi.getText().clear();
            edtPersonelSifre.getText().clear();
        }
    }
}

```

❖ Patent İşlemleri Sunumu

- Mirsis çalışanlarından Gökhan Beyin Patent işlemleriyle ilgili bir sunumu gerçekleştirdi.
- Patent, bir buluşun ya da buluşu kullanma hakkının kime ait olduğunu gösteren resmi bir belgedir.
- Patent alırken patent uzmanı bir avukatla beraber alınması daha sağlıklıdır. Çünkü patent işlemlerinde en küçük ayrıntılar bile çok önemlidir.
- Ayrıca kod veya algoritma için patent alınmaz.

KONTROL SONUCU:

YAPILAN İŞ: İnsan Kaynakları Semineri

❖ İnsan Kaynakları Semineri

- Mirsis İstanbul ofisinden İnsan Kaynakları personelleri bizlere sunum yapmak için geldiler. İşe alım süreçleri ve oryantasyon süreçleriyle ilgili ayrıntılı bilgiler verilen faydalı bir sunum oldu.
- İşe Alım Süreci
 - İş görüşmesinde nasıl davranmalıyız?
 - Görüşme esnasında rahatlık önemli midir?
 - GİRİM ve konuşma önemli midir?
 - CV hazırlanırken nelere dikkat edilmeli?
- Oryantasyon Süreci
 - İşe alındıktan sonraki süreçler.
 - Şirket kuralları, sözleşme, çalışma adabı bilgilendirmeleri.
 - Şirket hakkında bilgi toplama süreçleri.
 - Şirketin prensiplerini ve amaçlarını öğrenme süreçleri.



KONTROL SONUCU:

YAPILAN İŞ: Yönetici Login Web Servis ve Java

❖ Proje

- Projemizde bugün bazı planlamalar yaptık.
- Uygulayacağımız adımları sıraya koyduk.
- Daha önce yapmış olduğumuz “Yönetici Üyelik” kısmının devamı niteliğinde “Yönetici Login” işlemini gerçekleştirdik.

```
[WebMethod]
public int yoneticigiris(String kullanic Adi, String sifre)
{
    String vtKullaniciAdi = null, vtSifre = null;

    con = new SqlConnection("Data Source=mobilcuzdan.database.windows.net;Initial Catalog=mobilcuzdan;Integrated Security=False;User ID=
con.Open();
cmd = new SqlCommand("SELECT * FROM Yonetici WHERE kullanic_adi='" + kullanic Adi + "' AND sifre = '" + sifre + "'", con);

SqlDataReader reader = cmd.ExecuteReader();

if (reader != null)
{
    while (reader.Read())
    {
        vtKullaniciAdi = reader["kullanic_adi"].ToString().Trim();
        vtSifre = reader["sifre"].ToString().Trim();
    }
}

if (kullanic Adi.Equals(vtKullaniciAdi) && sifre.Equals(vtSifre))
{
    return 1;
}
else
{
    return 0;
}
}
```

```
public int yoneticigiris(String kullanic Adi, String sifre) {
    int sonuc = 0;

    SoapObject request = new SoapObject(NAMESPACE, METHOD_NAMES);
    request.addProperty("kullanic Adi", kullanic Adi);
    request.addProperty("sifre", sifre);

    SoapSerializationEnvelope envelope = new SoapSerializationEnvelope(SoapEnvelope.V1);
    envelope.dotNet = true;
    envelope.setOutputSoapObject(request);

    HttpTransportSE androidHttpTransportSE = new HttpTransportSE(URL);
    androidHttpTransportSE.debug = true;

    try {
        androidHttpTransportSE.call(SOAP_ACTIONS, envelope);

        SoapPrimitive response = (SoapPrimitive) envelope.getResponse();
        sonuc = Integer.parseInt(response.toString());
    } catch (Exception e) {
        e.printStackTrace();
    }

    return sonuc;
}
```

```
protected void onStart() {
    super.onStart();
    btGiris.setOnClickListener(new View.OnClickListener() {
        public void onClick(View v) {
            kullanic Adi = edtKullanic Adi.getText().toString();
            sifre = edtSifre.getText().toString();

            mAsyncTask mTask = new mAsyncTask();
            mTask.execute();
        }
    });
}

public class mAsyncTask extends AsyncTask<Void, Void, Void> {
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressBar.setVisibility(View.VISIBLE);
    }

    @Override
    protected Void doInBackground(Void... voids) {
        YoneticisWS object = new YoneticisWS();
        sonuc = object.yoneticigiris(kullanic Adi, sifre);
        return null;
    }

    @Override
    protected void onPostExecute(Void aVoid) {
        super.onPostExecute(aVoid);
        progressBar.setVisibility(View.GONE);
        if (sonuc == 1) {
            Intent intent = new Intent(LoginYoneticis.this, Yoneticis.class);
            intent.putExtra("KADI", kullanic Adi);
            intent.putExtra("SIFRE", sifre);
            startActivity(intent);
        }
    }
}
```

KONTROL SONUCU:

YAPILAN İŞ: Personel Ekle, Web Servis ve Java

❖ Proje

- Daha önce personelleri veritabanımıza kendimiz ekliyorduk.
- Bugün bu kısmı uygulama üzerinden ve ayrıca bir yönetici üzerinden yapabileceğimiz Web Servis ve Java kodlarımızı oluşturduk.
- Uygulamanın yeni durumuna göre yönetici kişi kendi hesabına girerek kendisine bağlı personelleri kullanıcı adı şifrelerini de belirleyerek sisteme üye edebiliyor.

```
[WebMethod]
public string PersonelEkleWs(String isim, String soyisim, String kullanıcıAdi, String sifre, int id, int limit, String tarih)
{
    int vtId = 0;

    con = new SqlConnection("Data Source=mobilcuzdan.database.windows.net;Initial Catalog=mobilcuzdan;Integrated Security=False;User
con.Open();

    cmd = new SqlCommand("INSERT INTO Personel (isim, soyisim, kullanıcı_adi, sifre, limit, harcama, bakiye, gıda, giyim, akaryakit,
diger, yönetici_id, c) VALUES ('" + isim + "', '" + soyisim + "', '" + kullanıcıAdi + "', '" + sifre + "', '" + limit + "'
, 0, 0, 0, 0, 0, 0, '" + id + "', '" + tarih + "')", con);

    int sonuc = cmd.ExecuteNonQuery();
    con.Close();

    if (sonuc > 0)
    {
        return "Personel eklendi";
    }
    else
    {
        return "Bu kullanıcı adı kullanılıyor";
    }
}
```

```
public String personelEkle(String isim, String soyisim, String kullanıcıAdi, String sifre, int id, int limit, String tarih)
{
    SoapObject request = new SoapObject(NAMESPACE, METHOD_NAME);
    request.addProperty("isim", isim);
    request.addProperty("soyisim", soyisim);
    request.addProperty("kullanıcıAdi", kullanıcıAdi);
    request.addProperty("sifre", sifre);
    request.addProperty("limit", limit);
    request.addProperty("id", id);
    request.addProperty("tarih", tarih);

    SoapSerializationEnvelope envelope = new SoapSerializationEnvelope(SoapEnvelope.VERSION1);
    envelope.doHeader = true;
    envelope.setOutputSoapObject(request);

    HttpTransportSE androidHttpTransportSE = new HttpTransportSE(URL);
    androidHttpTransportSE.debug = true;

    try {
        androidHttpTransportSE.call(SOAP_ACTION, envelope);
        SoapPrimitive response = (SoapPrimitive) envelope.getResponse();
        durum = response.toString();
    } catch (Exception e) {
        e.printStackTrace();
    }

    return durum;
}
```

```
protected void onStart() {
    super.onStart();
    btPersonelEkle.setOnClickListener((view) -> {
        final mAsyncTask mTask = new mAsyncTask();
        mTask.execute();
    });
}

public class mAsyncTask extends AsyncTask<Void, Void, Void> {
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressBar.setVisibility(View.VISIBLE);
        isim = edtIsim.getText().toString();
        soyisim = edtSoyisim.getText().toString();
        kullanıcıAdi = edtKullanıcıAdi.getText().toString();
        sifre = edtSifre.getText().toString();
        if (!edtLimit.getText().toString().equals("")) {
            limit = Integer.parseInt(edtLimit.getText().toString());
        }
    }

    @Override
    protected Void doInBackground(Void... voids) {
        if ((isim.equals("") || soyisim.equals("") || kullanıcıAdi.equals("") || sifre.equals("") || limit == 0 || limit < 0)) {
            uyari = 1;
            return null;
        } else {
            YoneticisWS yoneticisObject = new YoneticisWS();
            id = yoneticisObject.yoneticidGetir(yoneticikAdi);
        }
    }
}
```

- Yazılım Müdürümüz Ersin Beye haftalık raporumuzu yazılı bir şekilde e-mail yoluyla ilettik.
- Rapor içeriğine bu hafta neler yaptığımızı, hangi sorunlarla karşılaştığımızı ve gelecek hafta neler yapacağımızı yazdık.

KONTROL SONUCU:

YAPILAN İŞ: Personelleri Liste Şeklinde Görüntüleme, ListView

❖ Proje

- Yönetici tarafından eklenen personellerin görüntülenebilmesi için ListView görüntüleme modelini seçtik.
- Bu görüntüleme modeline göre eklenen personeller yukarıdan aşağıya sırayla liste biçiminde karşımıza çıkacak.

```

public AdaptorListView(Activity activity, ArrayList<YoneticiciDeger> personeller) {
    //XML'i alıp View'a çevirecek inflater'ı örnekleyelim
    mInflater = (LayoutInflater) activity.getSystemService(
        Context.LAYOUT_INFLATER_SERVICE);
    this.personeller = personeller;
}

@Override
public int getCount() { return personeller.size(); }

@Override
public YoneticiciDeger getItem(int position) { return personeller.get(position); }

@Override
public long getItemId(int position) { return position; }

@Override
public View getView(int position, View view, ViewGroup viewGroup) {
    View satirView;
    satirView = mInflater.inflate(R.layout.satir_listview, null);
    TextView ilkHarf=null;
    TextView isim=null;
    TextView soyisim=null;
    TextView bakiye=null;
    ilkHarf = (TextView) satirView.findViewById(R.id.satir_ilkHarf);
    isim = (TextView) satirView.findViewById(R.id.satir_isim);
    soyisim = (TextView) satirView.findViewById(R.id.satir_soyisim);
    bakiye = (TextView) satirView.findViewById(R.id.satir_bakiye);
    YoneticiciDeger pers;
    pers = personeller.get(position);
    //int i = pers.getPersonelIsim().indexOf(' ');
    String word = pers.getPersonelIsim().substring(0, 1).toUpperCase();

    ilkHarf.setText(word);
    isim.setText(pers.getPersonelIsim().trim());
    soyisim.setText(pers.getPersonelSoyisim().trim());
    bakiye.setText("₺" + pers.getPersonelBakiye());
    return satirView;
}

@Override
protected void onPause() {
    super.onPause();
    Intent intent2 = new Intent(TabPersoneller.this, TabPersonelEkle.class);
    intent2.putExtra("KADI", kullanciAdi);
}

public class mAsyncTask extends AsyncTask<Void, Void, Void> {
    private AdaptorListView adaptorListView;
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressBar.setVisibility(View.VISIBLE);
    }
    @Override
    protected Void doInBackground(Void... voids) {
        YoneticiciWS istek = new YoneticiciWS();
        personeller = istek.sonuc(kullanciAdi, sifre);
        while(true) {
            if(personeller != null) {
                adaptorListView = new AdaptorListView(TabPersoneller.this, personeller);
                return null;
            }
        }
    }
    @Override
    protected void onPostExecute(Void aVoid) {
        //super.onPostExecute(aVoid);
        listView.setAdapter(adaptorListView);
        listView.setDivider(null);
        listView.setDividerHeight(0);
        progressBar.setVisibility(View.GONE);
    }
}

```

KONTROL SONUCU:

❖ Proje

- ListView'a personel ekleme işleminden sonra sıra geldi bu personelleri yönetmeye.
- Listedeki personellerden herhangi birine tıklandığında o personelin harcama geçmişinin, bakiyesinin ve personele para gönderebileceğimiz bir EditText'in görüntülediği sayfa oluşturduk.
- Şimdilik sadece listeden tıklandığında personel sayfasına gitmesi için gerekenleri tamamladık. Çekmesi gereken personel bilgilerini ekledik.

```

public class mAsyncTask2 extends AsyncTask<Void, Void, Void> {
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressBar.setVisibility(View.VISIBLE);
    }
    @Override
    protected Void doInBackground(Void... voids) {
        YoneticisiWS istek = new YoneticisiWS();
        personeller = istek.sonuc(kullaniciAdi, sifre);
        return null;
    }
    @Override
    protected void onPostExecute(Void aVoid) {
        //super.onPostExecute(aVoid);
        progressBar.setVisibility(View.GONE);
    }
}

public void listViewDose() {
    final mAsyncTask mTask = new mAsyncTask();
    final mAsyncTask2 mTask2 = new mAsyncTask2();
    mTask.execute();
    listView.setOnItemClickListener((adapterView, view, i, l) -> {
        //mTask2.execute();
        try {
            //TimeUnit.SECONDS.sleep(5);
            ArrayList<String> strings = new ArrayList<>();
            ArrayList<Integer> persDeger = new ArrayList<>();
            strings.add(personeller.get(i).getPersonelIsim());
            strings.add(personeller.get(i).getPersonelKullaniciAdi());
            persDeger.add(personeller.get(i).getPersonelBakiye());
            persDeger.add(personeller.get(i).getPersonelGida());
            persDeger.add(personeller.get(i).getPersonelGiyim());
            persDeger.add(personeller.get(i).getPersonelAkaryakit());
            persDeger.add(personeller.get(i).getPersonelDiger());
            Intent intent = new Intent(TabPersoneller.this, Personelicerik.class);
            intent.putStringArrayListExtra("stringler", strings);
            intent.putIntegerArrayListExtra("harcamalar", persDeger);
            startActivity(intent);
        } catch (Exception e) {
            e.printStackTrace();
        }
    });
}

@Override
protected void onResume() {
    super.onResume();
    listViewDose();
}

```

KONTROL SONUCU:

YAPILAN İŞ: Proje Sunumları, Başarının Sırrı Semineri

❖ **Proje**

- Stajyerler olarak yürüttüğümüz projelerin gelinen noktasına kadarlık kısmının sunumunu yaptık.

❖ **Başarının Sırrı Semineri**

- Mirsis İnsan Kaynakları personellerinden Eda hanımın “Başarının Sırları” üzerine bir semineri gerçekleştirdi.
- Başarı **bir iş (çıktı)** için **gerekli olan enerjinin (motivasyon)** amaca uygun olarak **en etkili (mutluluk, tatmin, kapasite)** biçimde **kullanılmasıdır (performans)**.
- Küçük hedeflerden (resim), büyük resme geçmek (amaç).
- Beyin ve zihin farkları. Beyin zihin sistemini yönetir.
- Hedefler nasıl oluşturulur?
- İyi bir hedef daima olumlu olmalı. Beyin pozitif komut verirse olumlu şeyler hedeflenir. Olumlu düşünerek olumlu şeyler yaparız, hedefleriz.
- Hedeflerimizin başkalarına da faydalı olmasını sağlamalıyız. Bunu sağlarsak sosyal bir insan olarak daha rahat hedefe ulaşırız ve motivasyonumuz artar.
- Ulaşılabilir ve somut olmak gerekir. Amaca ulaşmak için muhakkak somut olmalıyız. “Ne olur da zengin olurum?” veya “Zengin olmak istiyorum!” doğru bir hedef değildir. Ancak “5000 liram olsun.” doğru bir hedeftir.
- Zihin yönetimi çok önemlidir. “Nerden geldim? Nereye gidiyorum?” burada zihin yönetimi devreye girer.
- Hedefimiz kontrolümüzde olmalı. 50000 TL’lik araba için 30000 TL’imiz var iken 20000 TL’ye ihtiyaç duyuyorsak bu 20000 TL’lik kısım kontrolümüzde olmayan kısım olduğundan bu nokta stres noktamız olacaktır. Bu durumda yapılacak iki şey var; Hedeften vazgeç? , Stresi kontrol et?
- Motivasyon nedir? (istekten türemiştir.) Motivasyon arttıkça iş yapma isteği artar.
- Kendimize sormalıyız. Gerçekten hedefe ihtiyacımız var mı? Popülerlik mesela? İstiyor muyuz? Gerçekten popüler mi olmalıyız? Gerek var mı?
- Eğer bir işte mutlu değilsek değer yargılarımızı gözden geçirmeliyiz. Bu performansı en etkili şekilde kullanmamız için önemlidir. Performans eyleme geçmiş motivasyondur.
- Düşün-Yap Egzersizi. Kendine komut ver, gerçekleştir.
- Karşılaştığımız problemleri pozitive çevirmeliyiz. Olumlu bakarak çözüm üretmeliyiz. Stres noktalarından uzaklaşmalıyız. Üzerine gitmeliyiz.



KONTROL SONUCU:

YAPILAN İŞ: Personel İçerik Sayfası Düzenleme, Personele Para Gönderme

❖ Proje

- Daha önce personelleri listelemek için bir ListView oluşturmuştuk ve listedeki herhangi bir personele tıklandığında yöneticiyi personel içerik sayfasına yönlendirmiştik.
- Bugün bu personel içerik sayfasını düzenlemeye başladık.
- Başlangıç olarak sayfaya ihtiyacımız olacak kadar TextView ve bir adet de EditText ekledik.
- İlk olarak personelin bakiyesine para ekleme/gönderme işlemini yapmaya karar verdik.

```
[WebMethod]
public String bakiyeVer(String kullanıcıAdi)
{
    con = new SqlConnection("Data Source=mobilcuzdan.database.windows.net;Initial Catalog=mobilcuzdan");
    con.Open();
    cmd = new SqlCommand("SELECT * FROM Personel WHERE kullanıcı_adi='" + kullanıcıAdi + "'", con);
    SqlDataReader reader = cmd.ExecuteReader();

    if (reader != null)
    {
        while (reader.Read())
        {
            while (i != 4)
            {
                degerler.Add(reader["bakiye"].ToString());
                i++;
            }
        }
    }
    con.Close();
    return degerler[3];
}
```

```
public int bakiyeGetir (String kullanıcıAdi) {
    int yeniBakiye=0;

    SoapObject request = new SoapObject("http://mobilcuzdan.azurewebsites.net/", "bakiyeVer");
    request.addProperty("kullanıcıAdi", kullanıcıAdi);

    SoapSerializationEnvelope envelope = new SoapSerializationEnvelope(SoapEnvelope.V1);
    envelope.dotNet = true;
    envelope.setOutputSoapObject(request);

    HttpTransportSE androidHttpTransportSE = new HttpTransportSE(URL);
    androidHttpTransportSE.debug = true;

    try {
        androidHttpTransportSE.call("http://mobilcuzdan.azurewebsites.net/bakiyeVer", envelope);

        SoapPrimitive response = (SoapPrimitive) envelope.getResponse();
        yeniBakiye = Integer.parseInt(response.toString());
    } catch (Exception e) {
        e.printStackTrace();
    }

    return yeniBakiye;
}
```

```
btnEkle.setOnClickListener((View) -> {
    mAsyncTask mTask = new mAsyncTask();
    mTask.execute();
});

public class mAsyncTask extends AsyncTask<Void, Void, Void> {
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressBar.setVisibility(View.VISIBLE);
        if (!edtEkle.getText().toString().equals("")) {
            bakiye = Integer.parseInt(edtEkle.getText().toString());
        } else {
            bakiye = 0;
        }
    }

    @Override
    protected void doInBackground(Void... voids) {
        DegerleriGonder gonder = new DegerleriGonder();
        YoneticisiWS bakiyeIste = new YoneticisiWS();
        if ((bakiye == 0) || (bakiye < 0)) {
            uyarı = 1;
            return null;
        }
        else {
            sonuc = gonder.paraEkle(isim, bakiye);
            bakiye2 = bakiyeIste.bakiyeGetir(kullanıcıAdi);
            uyarı = 2; // Toast Mesajı : "İŞLEM TAMAM"
            return null;
        }
    }

    @Override
    protected void onPostExecute(Void aVoid) {

```

KONTROL SONUCU:

YAPILAN İŞ: Personel İçerik Sayfası Düzenleme, Personel Silme

❖ Proje

- Personel içerik sayfamızı düzenlemeye devam ettik.
- Para Gönderme kısmımızdan sonra personeli silme işlemini yapmaya karar verdik.
- Personel Sil işlemi için personel içerik sayfasında bir adet Button kullandık.

```
[WebMethod]
public String personelSilWs(String kullanıcıAdi)
{
    String vtKullanıcıAdi = null;

    con = new SqlConnection("Data Source=mobilcuzdan.database.windows.net;Initial Catalog=mobilcuzdan;Integrated Sec
con.Open();

    cmd = new SqlCommand("SELECT kullanıcı_adi FROM Personel WHERE kullanıcı_adi = '" + kullanıcıAdi + "' ", con);
    SqlDataReader reader = cmd.ExecuteReader();

    if (reader != null)
    {
        while (reader.Read())
        {
            vtKullanıcıAdi = reader["kullanıcı_adi"].ToString();
        }
    }

    con.Close();

    if (vtKullanıcıAdi.Equals(kullanıcıAdi))
    {
        cmd2 = new SqlCommand("DELETE FROM Personel WHERE kullanıcı_adi = '" + kullanıcıAdi + "'", con);
        con.Open();
        cmd2.ExecuteNonQuery();
        con.Close();
        return "Personel silinmiştir";
    }

    return "Böyle bir personel yok";
}
```

```
public String personelSil(String kullanıcıAdi) {
    String durum = null;

    SoapObject request = new SoapObject(NAMESPACE, "personelSilWs");
    request.addProperty("kullanıcıAdi", kullanıcıAdi);

    SoapSerializationEnvelope envelope = new SoapSerializationEnvelope(SoapEnvelope.VERSION1);
    envelope.dotNet = true;
    envelope.setOutputSoapObject(request);

    HttpTransportSE androidHttpTransportSE = new HttpTransportSE(URL);
    androidHttpTransportSE.Debug = true;

    try {
        androidHttpTransportSE.call("http://blogprojasima.azurewebsites.net/personelSilWs", envelope);

        SoapPrimitive response = (SoapPrimitive) envelope.getResponse();
        durum = response.toString();
    } catch (Exception e) {
        e.printStackTrace();
    }

    return durum;
}
```

```
personelSil.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        AlertDialog.Builder builder = new AlertDialog.Builder(Personelicerik.this);
        builder.setTitle("Personel Silme");
        builder.setMessage("Personel silinecek!");
        builder.setNegativeButton("IPTAL", new DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog, int id) {
            }
        });
        builder.setPositiveButton("TAMAM", (dialog, id) -> {
            mAsyncTask2 mTask2 = new mAsyncTask2();
            mTask2.execute();
        });
        builder.show();
    }
});

public class mAsyncTask2 extends AsyncTask<Void, Void, Void> {
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressBar.setVisibility(View.VISIBLE);
    }
    @Override
    protected void doInBackground(Void... voids) {
        YoneticiWS object = new YoneticiWS();
        durum = object.personelSil(kullanıcıAdi);
        return null;
    }
    @Override
    protected void onPostExecute(Void aVoid) {
        super.onPostExecute(aVoid);
        progressBar.setVisibility(View.GONE);
        Toast.makeText(Personelicerik.this, durum, Toast.LENGTH_SHORT).show();
        finish();
    }
}
```

KONTROL SONUCU:

YAPILAN İŞ: Genel Toplantı, Modern Bankacılık Semineri

❖ Genel Toplantı

- Yazılım müdürümüz Ersin Bey ile stajyerler olarak olağan toplantı yapıldı.
- Yeni Fintech haberleriyle ilgili sohbet gerçekleştirildi.
 - Bitcoin değeri %15 düştü ve Microsoft desteğini çekti.
- Ersin Bey tarafından bankaların genel veritabanı sistemi üzerine bir anlatım yapıldı.
- Haftalık proje sunumlarımızı yaptık.

❖ Modern Bankacılık Semineri

- Akbank ve Mirsis bize modern bankacılığı anlatmak ve bizleri Fintech ile daha yakından tanıştırmak amacıyla iki Akbank yazılımcısıyla bir sunum gerçekleştirdi.
- Mobil bankacılık.
- Şube bankacılığının maliyetli olması ve en eski banka.
- İlk bankacılar : Tapınak Şövalyeleri
- Bankalar nasıl para kazanıyor?
 - Ucuza kaynak bul, pahalıya sat.
 - Komisyon.
- Bankalarda paralar gün sonunda merkez bankasında toplanır.
- 2001 ve 2008 krizleri.
- Rating şirketleri : Findeks
 - Eğer bir şirketin rating yani kredi notu iyiye daha düşük faizden kredi alabilir.
- Fiziksel paranın kaybolacağı dünya.
- BDDK (Bankacılık Düzenleme ve Denetleme Kurumu)
- Sadece tek bir ürüne odaklanma trendi.
 - Örn : Sadece kahve dükkanı, Nike'ın sadece birkaç model üzerinde geliştirmeler yapması.
- Atombank (Gelecek şube modeli)
- MBank (Sadece danışma görevi gören mini banka)



KONTROL SONUCU:

YAPILAN İŞ: Personel İçerik Sayfası Düzenleme, Personel Harcama Durumu

❖ Proje

- Personel içerik sayfamıza personele para gönderme ve personelin harcamalarını görüntüleme işlemleri asıl amacımızdı.
- Cuma günü Personel Sil işlemini de yaptıktan sonra bugün de Personel Harcama Durumu kısmımızı oluşturmaya başladık.
- İhtiyacımız kadar TextView ekleyerek Personel Harcama Durumu sayfamızı Web Servis ve Android kısmımızda programladık.

```

[WebMethod]
public String personelHarcama(String kullanıcıAdi, String tur, int harcanan, String tarih)
{
    int vtBakiye, vtHarcama, vtTur, sonuc, vtId=0;

    con = new SqlConnection("Data Source=mobilcudan.database.windows.net;Initial Catalog=mobilcudan;Integrated Security=False;User ID=sk;
    con.Open();

    SqlDataReader reader;
    switch (tur)
    {
        case "Gıda":
            cmd = new SqlCommand("SELECT Id, harcama, bakiye, gıda FROM Personel WHERE kullanıcı_adi ='" + kullanıcıAdi + "' ", con);
            reader = cmd.ExecuteReader();

            if (reader != null)
            {
                while (reader.Read())
                {
                    degerler.Add(reader["Id"].ToString());
                    degerler.Add(reader["harcama"].ToString());
                    degerler.Add(reader["bakiye"].ToString());
                    degerler.Add(reader["gıda"].ToString());
                }
            }
            break;
        case "Giyim":
            cmd = new SqlCommand("SELECT Id, harcama, bakiye, giyim FROM Personel WHERE kullanıcı_adi ='" + kullanıcıAdi + "' ", con);
            reader = cmd.ExecuteReader();

            if (reader != null)
            {
                while (reader.Read())
                {
                    degerler.Add(reader["Id"].ToString());
                    degerler.Add(reader["harcama"].ToString());
                    degerler.Add(reader["bakiye"].ToString());
                    degerler.Add(reader["giyim"].ToString());
                }
            }
            break;
        case "Akaryakit":
            cmd = new SqlCommand("SELECT Id, harcama, bakiye, akaryakit FROM Personel WHERE kullanıcı_adi ='" + kullanıcıAdi + "' ", con);
            reader = cmd.ExecuteReader();

            if (reader != null)
            {
                while (reader.Read())
                {
                    degerler.Add(reader["Id"].ToString());
                    degerler.Add(reader["harcama"].ToString());
                    degerler.Add(reader["bakiye"].ToString());
                    degerler.Add(reader["akaryakit"].ToString());
                }
            }
            break;
        case "Diğer":
            cmd = new SqlCommand("SELECT Id, harcama, bakiye, diger FROM Personel WHERE kullanıcı_adi ='" + kullanıcıAdi + "' ", con);
            reader = cmd.ExecuteReader();

            if (reader != null)
            {
                while (reader.Read())
                {
                    degerler.Add(reader["Id"].ToString());
                    degerler.Add(reader["harcama"].ToString());
                    degerler.Add(reader["bakiye"].ToString());
                    degerler.Add(reader["diger"].ToString());
                }
            }
            break;
    }

    return sonuc;
}

```

```

public String personelHarcama(String kullanıcıAdi, String tur, int harcanan, String tarih) {
    String sonuc = null;

    SoapObject request = new SoapObject(NAMESPACE, "personelHarcama");
    request.addProperty("kullaniciAdi", kullanıcıAdi);
    request.addProperty("tur", tur);
    request.addProperty("harcanan", harcanan);
    request.addProperty("tarih", tarih);

    SoapSerializationEnvelope envelope = new SoapSerializationEnvelope(SoapEnvelope.V1);
    envelope.dotNet = true;
    envelope.setOutputSoapObject(request);

    HttpTransportSE androidHttpTransportSE = new HttpTransportSE(URL);
    androidHttpTransportSE.debug = true;

    try {
        androidHttpTransportSE.call("http://blogprojesine.azurewebsites.net/personelHarcama", envelope);
        SoapPrimitive response = (SoapPrimitive) envelope.getResponse();
        sonuc = response.toString();
    } catch (Exception e) {
        e.printStackTrace();
    }

    return sonuc;
}

```

```

ArrayList<String> strings = new ArrayList<>();
ArrayList<Integer> persDeger = new ArrayList<>();

int bakiye, bakiye2, uyari;
String isim, kullanıcıAdi, durum, sonuc;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_personel_icerik);

    progressBar = (ProgressBar) findViewById(R.id.progressBar);

    Intent intent = getIntent();
    strings = intent.getStringArrayListExtra("stringler");
    persDeger = intent.getIntegerArrayListExtra("harcamalar");

    tvIsim = (TextView) findViewById(R.id.textView21);
    tvBakiye = (TextView) findViewById(R.id.textView68);
    tvGıda = (TextView) findViewById(R.id.textView23);
    tvGiyim = (TextView) findViewById(R.id.textView25);
    tvAkaryakit = (TextView) findViewById(R.id.textView27);
    tvDiger = (TextView) findViewById(R.id.textView29);
    btnEkle = (Button) findViewById(R.id.button7);
    btnSil = (Button) findViewById(R.id.btnPersonelSil);
    edtEkle = (EditText) findViewById(R.id.editText5);

    tvIsim.setText(strings.get(0));
    tvBakiye.setText(persDeger.get(0).toString());
    tvGıda.setText(persDeger.get(1).toString());
    tvGiyim.setText(persDeger.get(2).toString());
    tvAkaryakit.setText(persDeger.get(3).toString());
    tvDiger.setText(persDeger.get(4).toString());
    isim = strings.get(0);
    kullanıcıAdi = strings.get(1);
}

```

KONTROL SONUCU:

YAPILAN İŞ: Personel Harcama Sayfası

❖ Proje

- Yönetici kısmındaki işlemleri şimdilik bitirdikten sonra personel kısmına başladık.
- Personel kullanıcısı için bir harcama sayfası oluşturarak işe koyulduk
- Bir Spinner, bir EditText, bir de Button yardımıyla sayfamızı oluşturduk.
- Daha sonra Web Servis ve Android kısmında Personel harcama sayfamızı kodladık.

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_tab_harcama);

    Intent intent = getIntent();
    kullanciAdi = intent.getStringExtra("KADI");
    sifre = intent.getStringExtra("SIFRE");
    Tanimlar();
    SpinnerIslemi();

    btHarcamaKayit.setOnClickListener((view) -> {
        mAsyncTask mTask = new mAsyncTask();
        mTask.execute();
    });
}

@Override
protected void onStart() {
    super.onStart();
    mAsyncTask2 mTask2 = new mAsyncTask2();
    mTask2.execute();
}

public void Tanimlar() {...}

public void SpinnerIslemi(){
    harcamaSecenegi.setOnItemClickListener(new AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<?> adapterView, View view, int i, long l) {
            tur = harcamaSecenegi.getSelectedItem().toString();
        }
        @Override
        public void onNothingSelected(AdapterView<?> adapterView) {
        }
    });
}

public class mAsyncTask extends AsyncTask<Void, Void, Void> {
    @Override
    protected void onPreExecute() {
        super.onPreExecute();

        if(etHarcamaKayit.getText().toString().equals("")) {
            uyarı = 1;
        }else {
            harcanan = Integer.parseInt(etHarcamaKayit.getText().toString());
            if(!(harcanan == 0 || harcanan < 0)){
                uyarı = 2;
            }else{
                uyarı = 1; }}}

    @Override
    protected Void doInBackground(Void... voids) {
        if(uyarı == 2) {
            Tarih tarihObj = new Tarih();
            tarih = tarihObj.tarihVer();
            PersonelWS object = new PersonelWS();
            sonuc = object.personelHarcama(kullanciAdi, tur, harcanan, tarih);
            return null; }
        return null;}

    @Override
    protected void onPostExecute(Void aVoid) {
        super.onPostExecute(aVoid);
        etHarcamaKayit.getText().clear();
        if(uyarı == 1) {
            Toast.makeText(TabHarcama.this, "Lütfen geçerli bir değer giriniz.", Toast.LENGTH_SHORT).show();
        }else if(uyarı == 2) {
            Toast.makeText(TabHarcama.this, sonuc, Toast.LENGTH_LONG).show();
        }else {
            Toast.makeText(TabHarcama.this, "Bir hata oluştu.", Toast.LENGTH_SHORT).show(); }}}

```

KONTROL SONUCU:

YAPILAN İŞ: Personel Durum Sayfası, Finans Durumu

❖ Proje

- Personel bölümü için Personel Harcama kısmını tamamladıktan sonra sıra personelin kendi harcama durumunu, bakiyesini, harcama limitini görüntüleyeceği sayfaya geldi.
- Bunun için Personel bölümünün Personel Durum sayfasını oluşturmaya koyulduk.
- Gerektiği kadar TextView ekleyerek işe başladık.
- Ardından sayfaya personel için Finans Durumu bölmesi oluşturularak bu kısmı kodladık.

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_tab_durum);

    Intent intent = getIntent();
    kullanıcıAdi = intent.getStringExtra("KADI");
    sifre = intent.getStringExtra("SIFRE");

    tvLimit = (TextView) findViewById(R.id.textView52);
    tvHarcama = (TextView) findViewById(R.id.textView54);
    tvBakiye = (TextView) findViewById(R.id.textView56);
    tvIsim = (TextView) findViewById(R.id.textView49);
    cagirGelsin();
}
```

```
public class mAsyncTask extends AsyncTask<Void, Void, Void> {
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressBar.setVisibility(View.VISIBLE);
    }
    @Override
    protected Void doInBackground(Void... voids) {
        PersonelWS object = new PersonelWS();
        personelDurumu = object.sonuc(kullanıcıAdi, sifre);
        return null;
    }
    @Override
    protected void onPostExecute(Void aVoid) {
        super.onPostExecute(aVoid);
        tvLimit.setText(" " + personelDurumu.get(0).getPersonelLimit());
        tvHarcama.setText(" " + personelDurumu.get(0).getPersonelHarcama());
        tvBakiye.setText(" " + personelDurumu.get(0).getPersonelBakiye());
        tvIsim.setText(" " + personelDurumu.get(0).getPersonelIsim());
    }

    public void cagirGelsin() {
        mAsyncTask mTask = new mAsyncTask();
        mTask.execute();
    }

    @Override
    protected void onResume() {
        super.onResume();
        cagirGelsin();
    }
}
```

KONTROL SONUCU:

❖ Ece Hanım ve Nusret Beyin Ziyareti, Proje Sunumları ve Başarı Hikayesi

- Mirsis kurucularından Ece Hanım ve Mirsis Sistem Yöneticisi Nusret bey Teknopark ofisimizi ziyarete geldiler.
- Tanışma ve üzerinde çalıştığımız projeler üzerine sohbet gerçekleştirildi.
- Ersin Bey'e haftalık yaptığımız rapor sunumlarını bu sefer Ece Hanım ve Nusret Bey varken de yaptık.
- Ece Hanım bizlere Mirsis'i nasıl kurduğunu, hangi yollardan geçtiğini, bir nevi başarı hikayesini anlattı. Bu anlatı, karşılıklı soru-cevaplarla desteklendi.
- Ece Hanım ve Nusret Bey bizlere iş hayatına yönelik önerilerde bulundu. İş ahlakı ve iş etiğiyle ilgili tavsiyelerde bulunuldu.

❖ Sistem Yöneticiliği

- Mirsis Sistem Yöneticisi Nusret Bey bizlere Sistem Yöneticiliğiyle ilgili teknik bir sunum yaptı.
- Sistem yöneticiliğinin nasıl bir iş olduğu konusunda ve ilerde bu alanda uzmanlaşırsak ne gibi durumlarla karşılaşacağımız konusunda bilgiler verdi.
- Test-Analiz süreçleri.
- Sistemci ne yapar?

❖ Ersin Beyin Bilgisayar Mühendisi ve İş Hayatı Üzerine Sunumu

- Bilgisayar mühendisi ve iş hayatı
- Arz-talep ve arz-talep grafiği
- Türkiye'deki bilgisayar mühendisleri sayısı
- Bankacılık ve toplam istihdam
- Savunma Sanayii
- Türkiye Bilişim 500 zirvesi
- Dünyada çalışan insan başına ciro (silikon vadisi şirketleri)
- 2015 Yılı İş İlanları
 - %30 Yazılım >> “ %16 .Net, %9 Java, %5 Diğer”
- 2015 Yılı Bulunması Zor Uzmanlıklar
 - Android, iOS Programcılığı, Scrum Master, Java Developer, Digital Marketing, Full Stack Developer, CRM Developer

**KONTROL SONUCU:**

YAPILAN İŞ: Personel Durum Sayfası, Harcama Durumu

❖ Proje

- Daha önce personel durum sayfasını oluşturmuştuk.
- Önceden de belirlediğimiz gibi finans durumu kısmı ve harcama durumu kısmı bu durum sayfasında olacaktı.
- Finans durumunu tamamladıktan sonra personelin kendi harcama durumunu görüntüleyeceği harcama durumu kısmının yapımına başladık.
- Gerekli kadar TextView ekleyerek işe koyulduk.

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_tab_durum);

    Intent intent = getIntent();
    kullanıcıAdi = intent.getStringExtra("KADI");
    sifre = intent.getStringExtra("SIFRE");

    tvLimit = (TextView) findViewById(R.id.textView52);
    tvHarcama = (TextView) findViewById(R.id.textView54);
    tvBakiye = (TextView) findViewById(R.id.textView56);
    tvGida = (TextView) findViewById(R.id.textView61);
    tvGiyim = (TextView) findViewById(R.id.textView59);
    tvAkaryakit = (TextView) findViewById(R.id.textView63);
    tvDiger = (TextView) findViewById(R.id.textView65);
    tvIsim = (TextView) findViewById(R.id.textView49);
    cagirGelsin();
}

public class mAsynTask extends AsyncTask<Void, Void, Void> {
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressBar.setVisibility(View.VISIBLE);
    }
    @Override
    protected Void doInBackground(Void... voids) {
        PersonelWS object = new PersonelWS();
        personelDurumu = object.sonuc(kullanıcıAdi, sifre);
        return null;
    }
    @Override
    protected void onPostExecute(Void aVoid) {
        super.onPostExecute(aVoid);
        tvLimit.setText("" + personelDurumu.get(0).getPersonelLimit());
        tvHarcama.setText("" + personelDurumu.get(0).getPersonelHarcama());
        tvBakiye.setText("" + personelDurumu.get(0).getPersonelBakiye());
        tvGida.setText("" + personelDurumu.get(0).getPersonelGida());
        tvGiyim.setText("" + personelDurumu.get(0).getPersonelGiyim());
        tvAkaryakit.setText("" + personelDurumu.get(0).getPersonelAkaryakit());
        tvDiger.setText("" + personelDurumu.get(0).getPersonelDiger());
        tvIsim.setText("" + personelDurumu.get(0).getPersonelIsim());

        ArrayList<Entry> entries = new ArrayList<>();
        entries.add(new Entry(personelDurumu.get(0).getPersonelGida(), 0));
        entries.add(new Entry(personelDurumu.get(0).getPersonelGiyim(), 1));
        entries.add(new Entry(personelDurumu.get(0).getPersonelAkaryakit(), 2));
        entries.add(new Entry(personelDurumu.get(0).getPersonelDiger(), 3));

        ArrayList<String> labels = new ArrayList<>();
        labels.add("Gida");
        labels.add("Giyim");
        labels.add("Akaryakit");
        labels.add("Diğer");

        public void cagirGelsin() {
            mAsynTask mTask = new mAsynTask();
            mTask.execute();
        }

    @Override
    protected void onResume() {
        super.onResume();
        cagirGelsin();
    }
}

```

KONTROL SONUCU:

YAPILAN İŞ: Personel Harcama Geçmiş

❖ Proje

- Personel Harcama sayfası ve Personel Durum Görüntüleme sayfasını tamamladıktan sonra, personelimizin kendi harcama geçmişini görebileceği ayrı bir sayfa daha yapmaya karar verdik.
- Yeni bir tab ekledik ve ihtiyacımız kadar TextView ekledik.
- Daha sonra Web Servis ve Android tarafında bu kısmımızı kodladık.

```
[WebMethod]
public List<String> personelHarcamaGecmis(String kullanıcıAdi, String sifre)
{
    int personelId = 0;

    con = new SqlConnection("Data Source=mobilcuздan.database.windows.net;Initial Catalog=mobilcuздan;Integrated Security=False;User ID=
    con.Open();
    cmd = new SqlCommand("SELECT Id FROM Personel WHERE kullanıcı_adi ='" + kullanıcıAdi + "' AND sifre = '" + sifre + "'", con);

    SqlDataReader reader = cmd.ExecuteReader();
    if (reader != null)
    {
        while (reader.Read())
        {
            personelId = Convert.ToInt32(reader["Id"].ToString());
        }
    }
    con.Close();

    con.Open();
    cmd2 = new SqlCommand("SELECT harcama_turu, harcama_miktari, harcama_tarihi FROM Harcama_Raporu WHERE personel_id ='" + personelId +
    SqlDataReader reader2 = cmd2.ExecuteReader();
    if (reader2 != null)
    {
        while (reader2.Read())
        {
            degerler.Add(reader2["harcama_turu"].ToString());
            degerler.Add(reader2["harcama_miktari"].ToString());
            degerler.Add(reader2["harcama_tarihi"].ToString());
        }
    }
    con.Close();
    return degerler;
}
```

```
public ArrayList<HarcamaGecmisDegerler> harcamaGecmisi(String kullanıcıAdi, String sifre) {
    ArrayList<HarcamaGecmisDegerler> object = new ArrayList<>();

    SoapObject request = new SoapObject(NAMESPACE, "personelHarcamaGecmisi");
    request.addProperty("kullanıcıAdi", kullanıcıAdi);
    request.addProperty("sifre", sifre);

    SoapSerializationEnvelope envelope = new SoapSerializationEnvelope(SoapEnvelope.VENI);
    envelope.dotNet = true;
    envelope.setOutputSoapObject(request);

    HttpTransportSE androidHttpTransportSE = new HttpTransportSE(URL);
    androidHttpTransportSE.debug = true;

    try {
        androidHttpTransportSE.call("http://blogprojesine.azurewebsites.net/personelHarcamaGecmisi", envelope);

        SoapObject response = (SoapObject) envelope.getResponse();
        count = response.getPropertyCount();

        int i = 0, j = 0;

        for(i = 1; i <= count / 3; i++, j += 3) {

            object.add(new HarcamaGecmisDegerler(response.getPropertyAsString(i),
                                                    response.getPropertyAsString(i+1),
                                                    response.getPropertyAsString(i+2)));

        }
    } catch (Exception e) {
        e.printStackTrace();
    }

    return object;
}

public class mAsynTask extends AsyncTask<Void, Void, Void> {

    private AdapterGecmisListView adapterGecmisListView;

    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        progressBar.setVisibility(View.VISIBLE);
    }

    @Override
    protected void doInBackground(Void... voids) {

        PersonelWS gecmis = new PersonelWS();
        harcamaGecmisi = gecmis.harcamaGecmisi(kullanıcıAdi, sifre);

        while(true) {
            if(harcamaGecmisi != null) {
                adapterGecmisListView = new AdapterGecmisListView(TabGecmis.this, harcamaGecmisi);
                return null;
            }
        }

    }

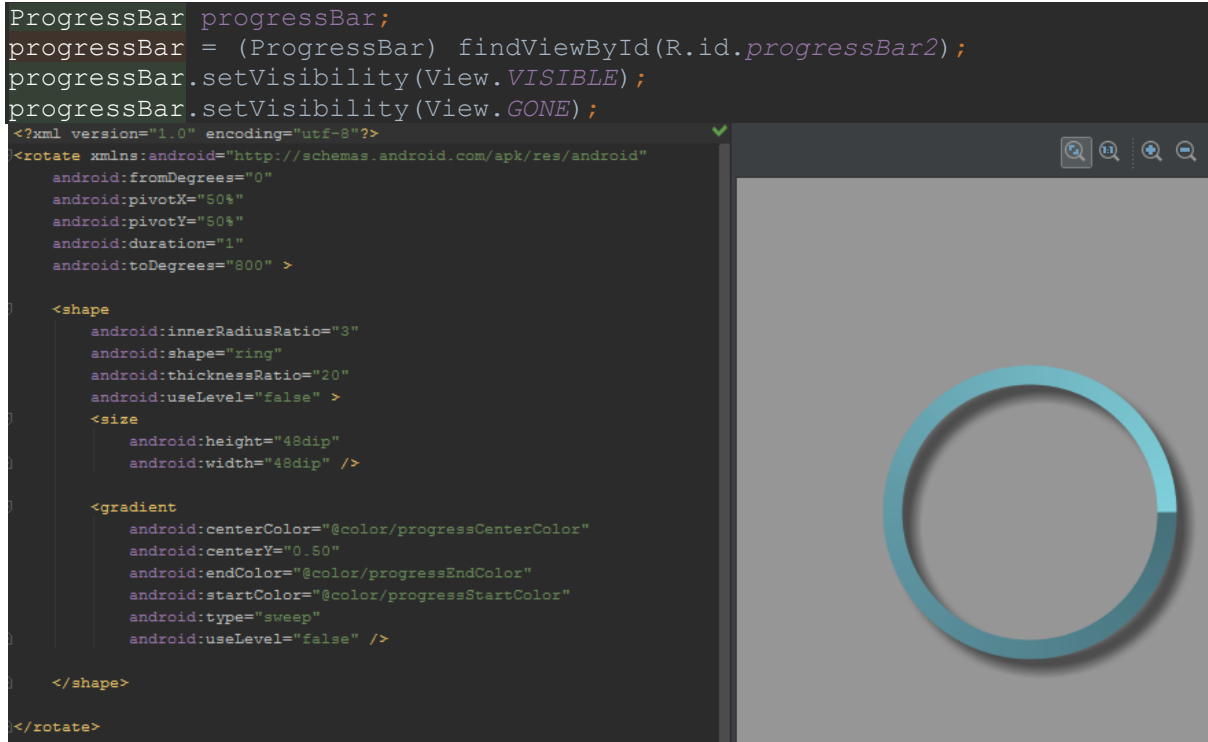
    @Override
    protected void onPostExecute(Void aVoid) {
        super.onPostExecute(aVoid);
        listView.setAdapter(adapterGecmisListView);
        listView.setDivider(null);
        listView.setDividerHeight(0);
        progressBar.setVisibility(View.GONE);
    }
}
```

KONTROL SONUCU:

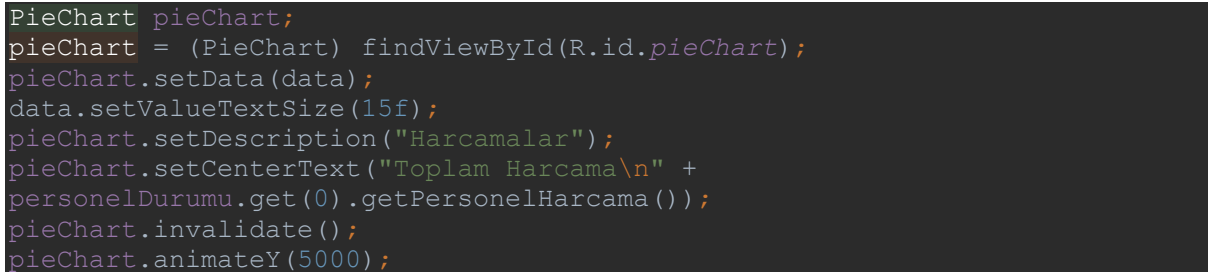
YAPILAN İŞ: ProgressBar ve Pasta Grafiği

❖ ProgressBar ve Pasta Grafiği

- Uygulamada sunucuya istek gönderdiğimiz bazı kısımlarda veritabanından bilgileri çekerken sayfanın bir süre beklemesi gerekiyordu. Bunu çözmek için o kısımlara bekleme komutları eklemiştik.
- Bekleme komutu kullanıcıya beklemesi gerektiğini göstermiyordu. Sadece sayfa boş bir şekilde bekliyordu. Bu durumu daha kullanıcı dostu yapmak için ProgressBar özelliğinden faydalanmak istedik.
- İnternet araştırmalarımız sonucunda bir ProgressBar tasarımı ve XML kodu bularak color, boyut gibi kısımlarını kendimize göre düzenledik ve uygulamada gereken yerlere aşağıda yazdığı gibi ekledik.



- Personelin kendi harcama durumunu görüntülediği sayfaya harcama durumlarının dağılımını gösteren bir pasta grafiği ekledik.



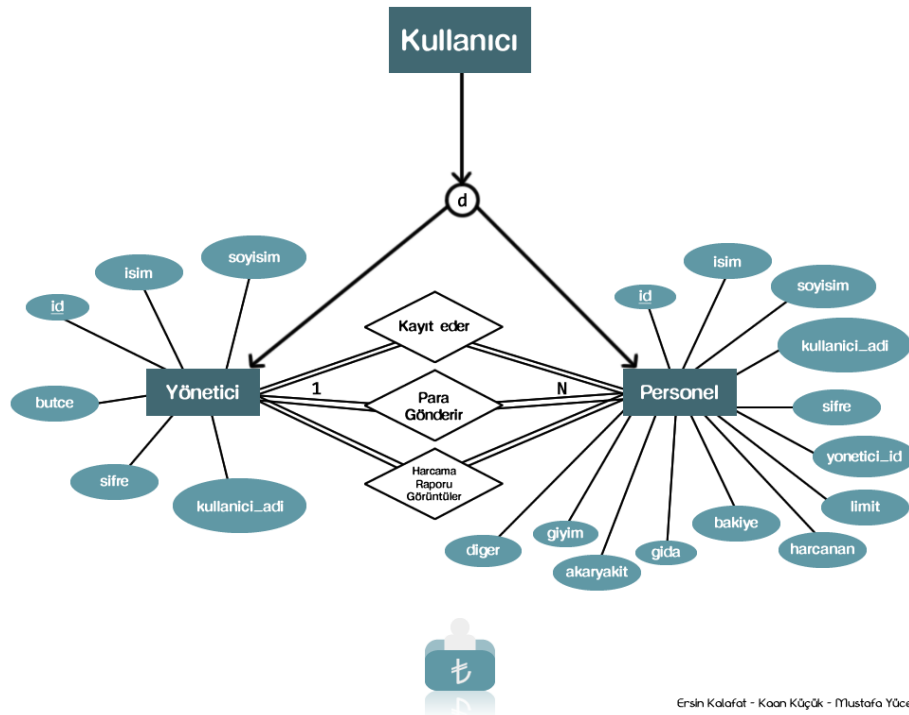
KONTROL SONUCU:

YAPILAN İŞ: Proje ER Şeması Oluşturma ve Kodların Ayrıntılı Düzenlenmesi

❖ Proje ER Şeması

- Yazılım müdürümüz Ersin Bey, yaptığımız projelerin ER diyagramlarını ne şekilde oluşturduğumuzu görmek ve bizim yaptığımız bir projenin ER diyagramını nasıl çıkaracağımızı tecrübe etmemiz için gruptan ER diyagramı çizmesini istedi.
- Biz de kendi proje grubumuz olarak projemizin ER diyagramını çıkardık.

ucluPriz - Personel-Cüzdan ER Diyagramı



Ersin Kalafat - Kaan Küçük - Mustafa Yücel

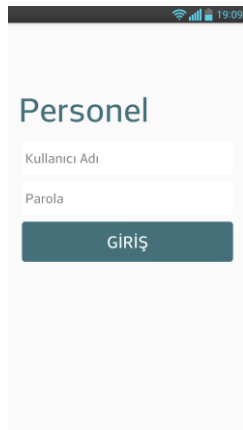
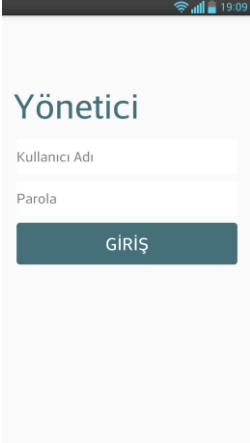
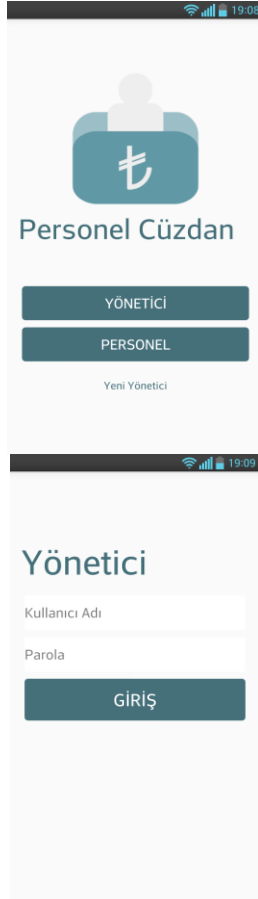
❖ Kod Düzenlemeleri

- Yönetici Tarafı**
 - Yönetici girişi için hata kontrolü (Kullanıcı Adı ve Şifre)
 - Yönetici para gönderirken EditText'i boş bıraktığında hata alınması işlemi
 - Personel eklenirken EditText'leri boş bırakınca hata vermesi işlemi
 - Login ekranında şifre girilen EditText'in **** şeklinde password girişi olarak değiştirilmesi
 - Personel eklerken limit değeri sorulması işlemi
- Personel Tarafı**
 - Personel girişi için hata kontrolü (Kullanıcı Adı ve Şifre)
 - Harcama yapmadan kayıt butonuna basılırsa hata verme işlemi
- Yönetici Üyelik**
 - Yönetici üye olurken alanları boş bırakması durumunda hata verdirilme işlemi

KONTROL SONUCU:

YAPILAN İŞ: XML Dizayn : İntro, Yönetici Login, Personel Login

❖ XML Dizayn : Intro, Yönetici Login, Personel Login



```
<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Yönetici"
    android:id="@+id/button"
    android:layout_above="@+id/button2"
    android:layout_alignParentLeft="false"
    android:background="@drawable/buton_tanimlama"
    android:layout_margin="5dp"
    android:textColor="@#ffffff"
    android:textSize="20dp" />

<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Personel"
    android:id="@+id/button2"
    android:layout_alignParentLeft="true"
    android:background="@drawable/buton_tanimlama"
    android:layout_margin="5dp"
    android:textSize="20dp"
    android:textColor="@#ffffff" />

<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Yeni Yönetici"
    android:id="@+id/yoneticimiis"
    android:layout_alignParentLeft="true"
    style="@android:attr/borderlessButtonStyle"
    android:layout_margin="5dp"
```


```
<LinearLayout
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="80dp"
    android:id="@+id/linearlayout2">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Kullanıcı Adı"
        android:id="@+id/textview20"
        android:textSize="20dp"
        android:textColor="@color/textviewRenk"
        android:textIsSelectable="false"
        android:layout_marginBottom="10dp" />
    <EditText
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:id="@+id/editText13"
        android:visibility="visible"
        android:hint="Kullanıcı Adı"
        android:background="@color/editTextRenk"
        android:textSize="20dp"
        android:textAlignment="inherit"
        android:layout_margin="5dp"
        android:inputType="text"
        android:paddingLeft="5dp" />
    <EditText
        android:layout_width="match_parent"
        android:layout_height="50dp"
```

```
        android:id="@+id/editText6"
        android:visibility="visible"
        android:hint="Kullanıcı Adı"
        android:background="@color/editTextRenk"
        android:textSize="20dp"
        android:textAlignment="inherit"
        android:layout_margin="5dp"
        android:inputType="text"/>
    <EditText
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:id="@+id/editText7"
        android:hint="Parola"
        android:background="@color/editTextRenk"
        android:textSize="20dp"
        android:textAlignment="inherit"
        android:layout_margin="5dp"
        android:inputType="textPassword" />
    <Button
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:text="Giriş"
        android:background="@drawable/buton_tanimlama"
        android:id="@+id/button3"
        android:layout_gravity="center_horizontal"
        android:layout_margin="5dp"
        android:textSize="20dp"
        android:textColor="@#ffffff"
        android:layout_weight="0.04"/>
```

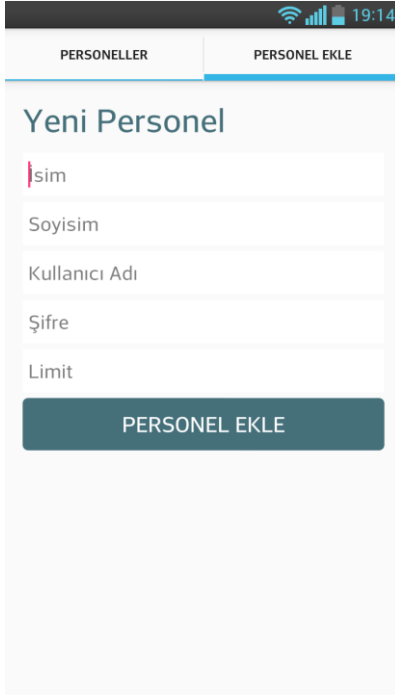
KONTROL SONUCU:

YAPILAN İŞ: XML Dizayn : Yönetici Üyelik, Personel Ekleme

❖ XML Dizayn : Yönetici Üyelik, Personel Ekleme



```
<LinearLayout
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/linearLayout14">
    <LinearLayout
        android:orientation="horizontal"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginTop="50dp"
        android:layout_gravity="center_vertical">
        <EditText
            android:layout_width="match_parent"
            android:layout_height="40dp"
            android:id="@+id/edtYEkleIsim"
            android:layout_gravity="center_vertical"
            android:background="@color/editTextRenk"
            android:hint="İsim"
            android:paddingLeft="5dp" />
    </LinearLayout>
    <LinearLayout
        android:orientation="horizontal"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginTop="5dp">
        <EditText
            android:layout_width="match_parent"
            android:layout_height="40dp"
            android:id="@+id/edtYEkleSoyisim"
            android:layout_gravity="center_vertical"
            android:background="@color/editTextRenk"
            android:paddingLeft="5dp" />
    </LinearLayout>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Yeni Personel"
        android:id="@+id/textView66"
        android:textColor="@color/textViewRenk"
        android:textSize="30dp" />
    <LinearLayout
        android:orientation="vertical"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/linearLayout14">
        <LinearLayout
            android:orientation="horizontal"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_marginTop="50dp"
            android:layout_gravity="center_vertical">
            <EditText
                android:layout_width="match_parent"
                android:layout_height="40dp"
                android:id="@+id/editText9"
                android:layout_gravity="center_vertical"
                android:background="@color/editTextRenk"
                android:hint="İsim"
                android:paddingLeft="5dp" />
        </LinearLayout>
        <LinearLayout
            android:orientation="horizontal"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_marginTop="5dp">
            <EditText
                android:layout_width="match_parent"
                android:layout_height="40dp"
                android:id="@+id/editText9"
                android:layout_gravity="center_vertical"
                android:background="@color/editTextRenk"
                android:paddingLeft="5dp" />
            </LinearLayout>
        </LinearLayout>
    </LinearLayout>
```



```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Yeni Personel"
    android:id="@+id/textView66"
    android:textColor="@color/textViewRenk"
    android:textSize="30dp" />
    <LinearLayout
        android:orientation="vertical"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/linearLayout14">
        <LinearLayout
            android:orientation="horizontal"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_marginTop="50dp"
            android:layout_gravity="center_vertical">
            <EditText
                android:layout_width="match_parent"
                android:layout_height="40dp"
                android:id="@+id/editText9"
                android:layout_gravity="center_vertical"
                android:background="@color/editTextRenk"
                android:hint="İsim"
                android:paddingLeft="5dp" />
        </LinearLayout>
        <LinearLayout
            android:orientation="horizontal"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_marginTop="5dp">
            <EditText
                android:layout_width="match_parent"
                android:layout_height="40dp"
                android:id="@+id/editText9"
                android:layout_gravity="center_vertical"
                android:background="@color/editTextRenk"
                android:paddingLeft="5dp" />
            </LinearLayout>
        </LinearLayout>
    </LinearLayout>
```

KONTROL SONUCU:

YAPILAN İŞ: XML Dizayn : Personel İçerik

❖ XML Dizayn : Personel İçerik

Mehmet

Gıda : : 120 TL

Giyim : : 65 TL

Akaryakıt : : 150 TL

Diğer : : 40 TL

Bakiye : 2125 TL

Para Ekle

PARA GÖNDER

PERSONELİ SİL

```

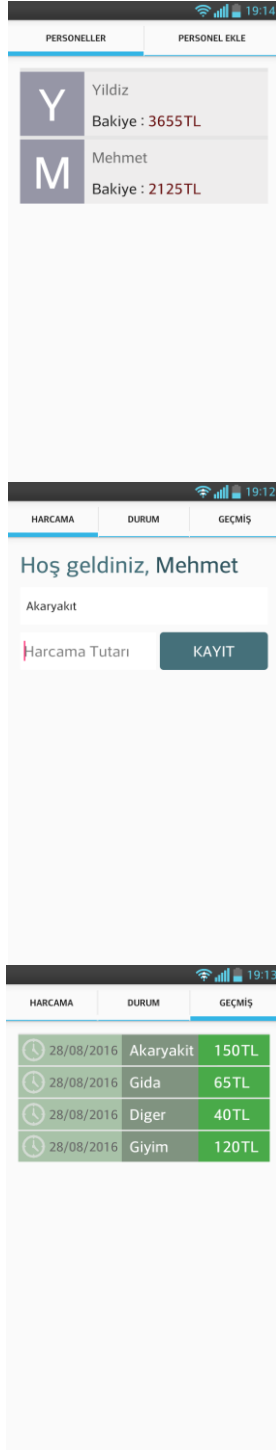
<LinearLayout
    android:id="@+id/linearLayout4"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">
    <TextView
        android:id="@+id/textView21"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="- Personel İsmi -"
        android:textColor="@color/butonSecildiginde"
        android:textSize="25dp" />
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:layout_marginTop="10dp">
        <LinearLayout
            android:orientation="horizontal"
            android:layout_width="match_parent"
            android:layout_height="match_parent">
            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Bakiye"
                android:id="@+id/textView67"
                android:textSize="20dp"
                android:textColor="@color/textViewRenk" />
            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:orientation="horizontal"
                android:layout_width="match_parent"
                android:layout_height="match_parent"
                android:layout_marginTop="20dp">
                <EditText
                    android:layout_width="180dp"
                    android:layout_height="50dp"
                    android:id="@+id/editText5"
                    android:textSize="20dp"
                    android:background="@color/white"
                    android:layout_gravity="center_vertical"
                    android:hint="Para Ekle"
                    android:inputType="number"/>
                <Button
                    android:layout_width="match_parent"
                    android:layout_height="50dp"
                    android:text="Para Gönder"
                    android:id="@+id/button7"
                    android:background="@drawable/buton_tanimlama"
                    android:layout_marginLeft="5dp"
                    android:textColor="@color/white"
                    android:textSize="20dp" />
            </LinearLayout>
        <LinearLayout
            android:orientation="horizontal"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_marginTop="180dp">
            <Button
                android:layout_width="match_parent"
                android:layout_height="50dp"
                android:text="Personeli Sil"
                android:id="@+id/btnPersonelSil"
                android:background="@drawable/buton_tanimlama"
                android:layout_gravity="bottom"
                android:layout_marginLeft="5dp"
                android:textColor="@color/white"
                android:textSize="20dp" />
            </LinearLayout>
    </LinearLayout>

```

KONTROL SONUCU:

YAPILAN İŞ: XML Dizayn : Personel Listesi, Harcama Sayfası, Geçmiş

❖ XML Dizayn : Personel Listesi, Harcama Sayfası, Geçmiş



```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="16dp"
    android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    tools:context="com.example.ersinkalafat.temmuz14.TabPersoneller">
    <ListView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/listView">
    </ListView>
    <ProgressBar
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/progressBar2"
        android:layout_centerVertical="true"
        android:layout_centerHorizontal="true"
        style="?android:attr/progressBarStyleLarge"
        android:visibility="gone"
        android:indeterminateDrawable="@drawable/progress"
        />
</RelativeLayout>
```

```
<LinearLayout
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/linearlayout8"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="10dp">
    <Spinner
        android:layout_width="match_parent"
        android:layout_height="50dp"
        android:id="@+id/spinner2"
        android:background="@#ffffff" />
    <LinearLayout
        android:orientation="horizontal"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginTop="10dp">
        <EditText
            android:layout_width="180dp"
            android:layout_height="50dp"
            android:id="@+id/editText8"
            android:textSize="16dp"
            android:background="@color/editTextRenk"
            android:layout_gravity="center_vertical"
            android:hint="Harcama Tutarı"
            android:inputType="number"
            android:paddingLeft="5dp" />
        <Button
            android:layout_width="match_parent"
            android:layout_height="50dp"
            android:text="Kayıt" />
    </LinearLayout>
</LinearLayout>
```

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="16dp"
    android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:paddingTop="16dp"
    tools:context="com.example.ersinkalafat.temmuz14.TabGecmis">
    <ListView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/gecmisListView">
    </ListView>
    <ProgressBar
        style="?android:attr/progressBarStyleLarge"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/progressBarGecmis"
        android:layout_centerVertical="true"
        android:layout_centerHorizontal="true"
        android:visibility="gone"
        android:indeterminateDrawable="@drawable/progress"/>
</RelativeLayout>
```

KONTROL SONUCU:

❖ XML Dizayn : Personel Finans ve Harcama Durumları Sayfası



```

<LinearLayout
    android:orientation="horizontal"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/linearlayout8"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="20dp"
    android:id="@+id/linearlayout9">
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Finans Durumu"
        android:id="@+id/textview50"
        android:textColor="@color/butonSecildiginde"
        android:textSize="25dp" />
    </LinearLayout>
    <LinearLayout
        android:orientation="horizontal"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/linearlayout9"
        android:layout_centerHorizontal="true"
        android:id="@+id/linearlayout10"
        android:layout_marginTop="20dp"
        android:layout_marginLeft="10dp">
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Harcama Durumu"
            android:id="@+id/textview51"
            android:textColor="@color/butonSecildiginde"
            android:textSize="25dp" />
        </LinearLayout>
        <LinearLayout
            android:id="@+id/linearlayout4"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:layout_below="@+id/linearlayout10"
            android:layout_marginTop="20dp"
            android:layout_marginLeft="10dp">
            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:orientation="horizontal">
                <LinearLayout
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_marginTop="5dp"
                    android:textColor="@color/butonSecildiginde"
                    android:textSize="25dp" />
                <TextView
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:text="Gıda"
                    android:id="@+id/textview10"
                    android:layout_marginTop="5dp"
                    android:textColor="@color/butonSecildiginde"
                    android:textSize="25dp" />
                <TextView
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:text="Giyim"
                    android:id="@+id/textview11"
                    android:layout_marginTop="5dp"
                    android:textColor="@color/butonSecildiginde"
                    android:textSize="25dp" />
            </LinearLayout>
            <LinearLayout
                android:orientation="vertical"
                android:layout_width="385dp"
                android:layout_height="385dp"
                android:layout_marginRight="10dp">
                <com.github.mikephil.charting.charts.PieChart
                    android:id="@+id/pieChart"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"/>
            </LinearLayout>
        </LinearLayout>
    </LinearLayout>

```

KONTROL SONUCU:

YAPILAN İŞ: Sunum Oluşturma

❖ Sunum

- Projemiz hedeflendiği gibi tamamlandıktan sonra Teknopark müdürleri ve Mirsis Müdürlerinin katılacağı bir sunum gerçekleştirileceğimiz söylendi.
- Gerçekleştirilecek sunum için bir Powerpoint sunusu oluşturdum.



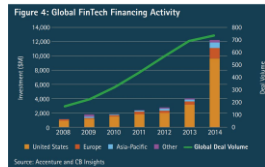
Fintech

Başka bir deyişle, teknolojinin yardımıyla, finansal hizmetleri daha kullanışlı hale getirerek finansal hizmetlere erişimin yeni yollarla sağlanmasıdır



Fintech

Türkiye'de Fintech pazarının 20 Milyar Dolara koştuğu ve 2016 yılının Fintech yılı olacağı söyleniyor



Bizim Projemiz

Yeni bir iş uygulaması



Fintech

Fintech, finans ile teknolojinin birleşiminden türeyen yeni bir akımdır



Fintech

Fintech'i son yıllarda gündeme getiren unsurların başında internet kullanımı ve mobil uygulamalardaki hızlı artış geliyor



Neden Fintech?

Verilerden de anlaşıldığı gibi Fintech alanında bizleri büyük bir pazar bekliyor. Bu pazarda yer almak hem şimdi için hem de gelecek için ciddi önem taşıyor. Biz de buna inanarak bu pazarda bir yer edinmek istiyoruz



Bizim Projemiz

Projemizin amacı, genellikle orta ve alt seviye şirketlerde uygulanan bir yöntemi basite indirmek



