

12.04.2020

Kişisel Sağlık İzleme

Sağlık Dostu(Healt Friendly)

Rapor - 2

**AYŞE ÇAY, AYŞE UYAR, CANKAT ATEŞ, RECEP BATUHAN GÜNAY,
RÜMEYSA TAŞ**
YOZGAT BOZOK ÜNİVERSİTESİ

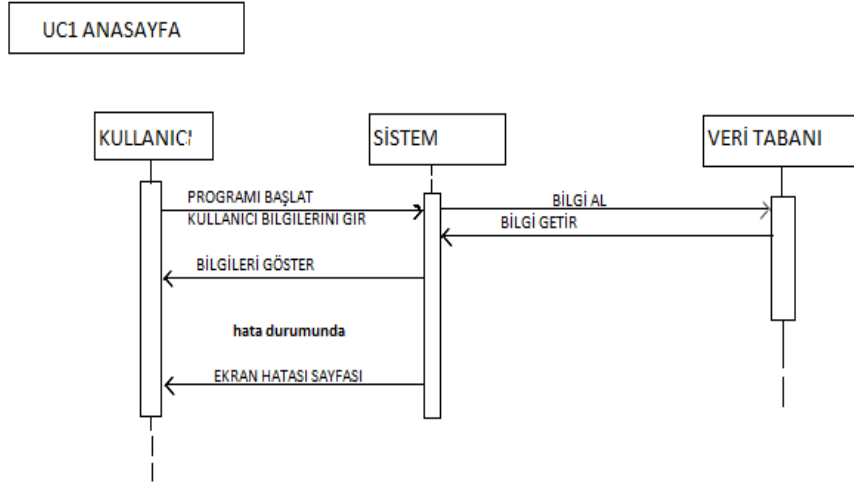
İçindekiler

Katkı Dağılımı Tablosu	2
1-) Etkileşim Diyagramları.....	3
2-)SINIF DİYAGRAMI VE ARAYÜZ ÖZELLİKLERİ	12
A. Sınıf Diyagramı	12
B. Veri Türleri ve İşlem İşaretleri	12
C. İZLENEBİLİRLİK MATRİSİ	15
D. NESNE KISITLAMA DİLİ (OCL) SÖZLEŞMELERİ	16
3-)SİSTEM MİMARİSİ VE SİSTEM TASARIMI	18
A. MİMARİ STİLLER	18
B. Alt Sistemlerin Tanımlanması	19
C. Alt Sistemlerin Donanıma İşlemi.....	20
D. Kalıcı Veri Depolama.....	20
E. Ağ Protokolleri.....	21
F. GLOBAL KONTROL AKIŞI	21
D. DONANIM GEREKSİNİMLERİ	22
4-)Algoritmalar ve Veri Yapıları.....	22
A-)ALGORİTMALAR	22
Algoritmanın örnek kod gösterimi :.....	25
Adıma göre yakılan kalori:	30
B.VERİ YAPILARI.....	32
5-) Kullanıcı Arayüz Tasarımı ve Uygulaması	32
6-)Testlerin Tasarımı.....	40
A. Sınıf Testleri	40
7-)Proje Yönetimi ve Çalışma Planı.....	42
b.Proje Koordinasyon ve İlerleme Raporu.....	43
d. Sorumlulukların Dağılımı	44
8-) Kaynakça	45

Katkı Dağılımı Tablosu

Görev	Recep Batuhan Günay	Cankat Ateş	Ayşe Uyar	Ayşe Çay	Rümeysa Taş	Toplam
Etkileşim Diyagramları	50	35	5	5	5	100
Sınıf Diyagramları ve Arayüz Özellikleri	5	5	40	40	5	100
Sistem Mimarisi ve Sistem Tasarımı	5	5	40	40	5	100
Algoritmalar ve Veri Yapıları	5	5	5	5	80	100
Kullanıcı Arayüzü Tasarımı ve Uygulaması	20	20	20	20	20	100
Test Tasarımı	20	20	20	20	20	100
Proje Yönetimi ve Çalışma Planı	20	20	20	20	20	100
Kaynakça	20	20	20	20	20	100

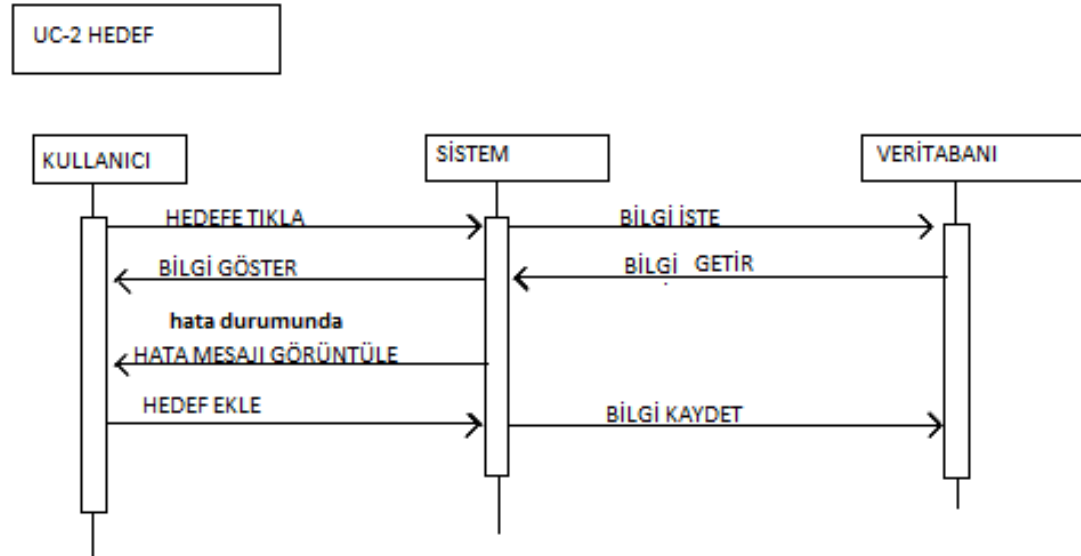
1-) Etkileşim Diyagramları



DİYAGRAM-1 UC1 ANASAYFA

Diyagram-1 “UC-1 ANASAYFA” Use Case 1’in dizi diyagramı gösterilmektedir. Kullanıcı bilgilerini almak ve kullanıcı girişi için kullanılır. Başlatıcı aktörü kullanıcıdır. Aktörün (Yani kullanıcının) hedefi genel sağlık durumuna erişmek ve takip etmektir. Yardımcı aktör veri tabanıdır. Önkoşul, uygulamanın giriş sayfasında görüntülenmesidir. Beklenen koşulu, kullanıcıların sağlıkları hakkında bilgilere erişmek için kullanılan panel olmasıdır. Başarılı senaryo için gidilecek yol:

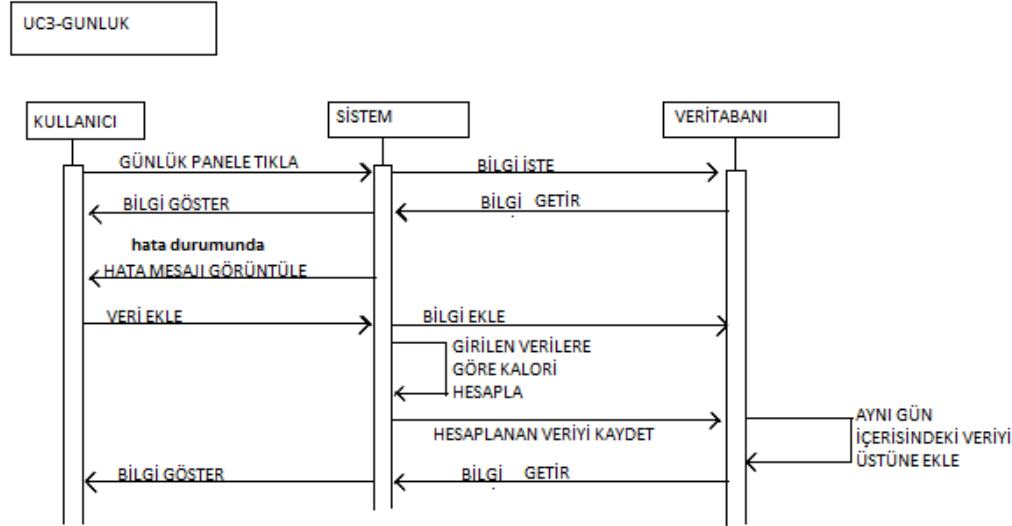
İlk olarak kullanıcı programı başlatır. Kullanıcı bilgilerini girer. Sistem veri tabanından bilgi alır ve sisteme bilgi getirir. Kullanıcıya bilgilerini gösterir. Aksi takdirde sistem kullanıcıya ekran hata sayfası açar, bu durumda kullanıcının tekrar denemesi gerekir.



DIYAGRAM-2 UC2 HEDEF

Diyagram-2 “UC2 HEDEF” Use Case 2’nin dizi diyagramı gösterilmektedir. Kullanıcının günlük hedeflerini amaçlamaktadır. Başlatıcı aktör kullanıcıdır. Aktörün (yani kullanıcının) hedefi, kendine belirlediği hedefe ilerleme durumunu takip etmektir. Katılan aktör, veri tabanıdır. Önkoşulu, kullanıcı ana sayfaya giriş yaptıktan sonra panele tıklanması gereklidir. Beklenen şartı, kullanıcının kendine belirlemiş olduğu hedefleri vekadyettiği ilerlemeyi göstermektir. Başarılı senaryo için gidilecek yol:

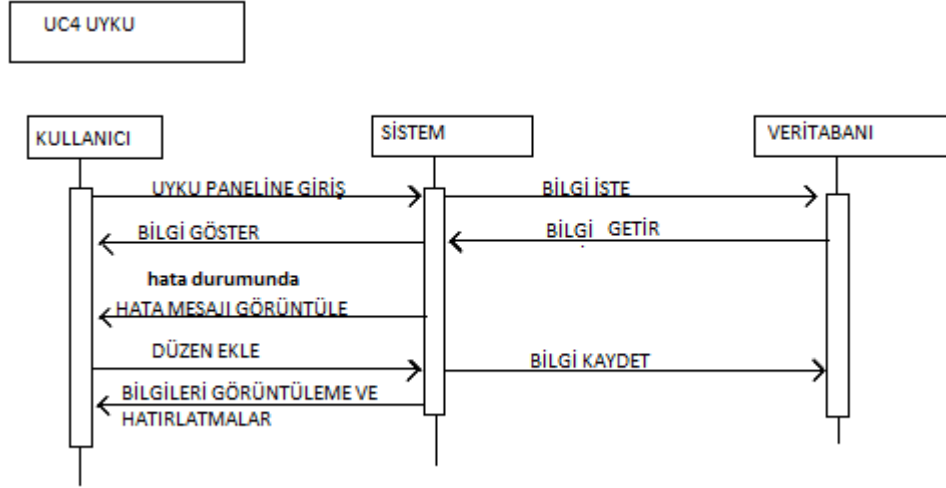
- 1.Kullanıcı uygulamaya giriş yapar
- 2.Kullanıcı hedef paneline giriş yapar
- 3.Sistem veri tabanından bilgi ister
- 4.Veri tabanı bilgiyi getirir
- 5.Kullanıcı bilgileri görür
- Olası bir hata durumunda
- 6.Veri tabanında kayıtlı bir bilgi (kullanıcının hedefi) yoktur.
- 7.Kullanıcı bilgi (Hedef) ekler.



DIYAGRAM-3 UC3 GÜNLÜK

Diyagram-3 “UC-3 GÜNLÜK” Use-Case 3’ün dizi diyagramı gösterilmektedir. Günlük alınan verileri gösterilmesi ve hesaplanması için kullanılır. Başlatıcı aktörü kullanıcıdır. Aktörün hedefi kullanıcının günlük değerlerini ve aktivite durumunu gösterilmesidir. Yardımcı aktör veri tabanıdır. Önkoşulu, kullanıcının ana sayfaya girdikten sonra günlük paneline de giriş yapması gerekmektedir. Beklenen şart, gün içerisinde alınan ve kaydedilen kalorilerin hesabını, vücudumuza aldığımız su miktarını, kullanıcının gün içerisindeki öğünlerde aldığı kalorileri kaydetmektir ve bunları kullanıcıya göstermektir. Başarılı bir senaryo için gidilecek yol:

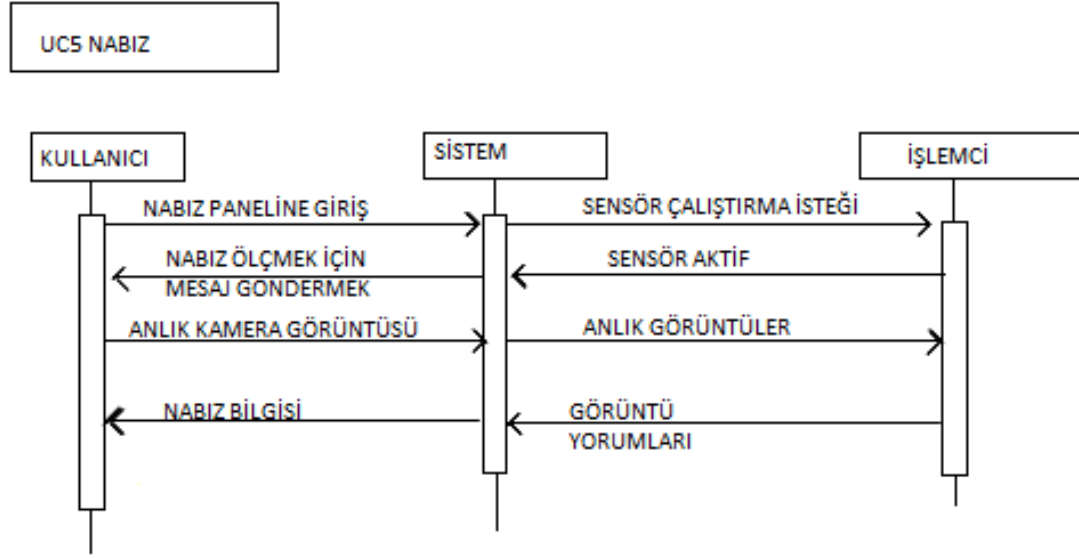
- 1.Kullanıcı uygulamaya giriş yapar
- 2.Kullanıcı günlük paneline giriş yapar
- 3.Sistem veri tabanından bilgi ister
- 4.Bilgiyi getirir
- 5.Bilgiyi gösterir
- 6.Kullanıcı veri ekler
- 7.Sistem veri tabanına bilgiyi ekler
- 8.Eklenen verilere göre tekrar hesaplanmalar yapılır.
- 9.Hesaplanan veriyi veri tabanına kaydeder.
- 10.Hesaplanan veri kullanıcıya gösterilir.



DIYAGRAM-4 UC4 UYKU

Diyagram-4 “UC-4 UYKU” Use-Case 4’ün dizi diyagramı gösterilmektedir. Kullanıcının uyku düzenini gösterir ve hatırlatmalarda bulunduğu use case’dir. Başlatıcı aktör kullanıcıdır ve aktörün (kullanıcının) hedefi uyku düzenini sağlamaktır. Yardımcı (katılan) aktör ver tabanıdır. Önkoşulları , kullanıcının ana sayfa bölümünden uyku paneline giriş yapması gerekmektedir. Beklenen şart, kullanıcının kendisi belirlediği uyku düzenini kullanıcıya sağlamak ve ona hatırlatmalarda bulunmaktır. Başarılı bir senaryo için gidilecek yol:

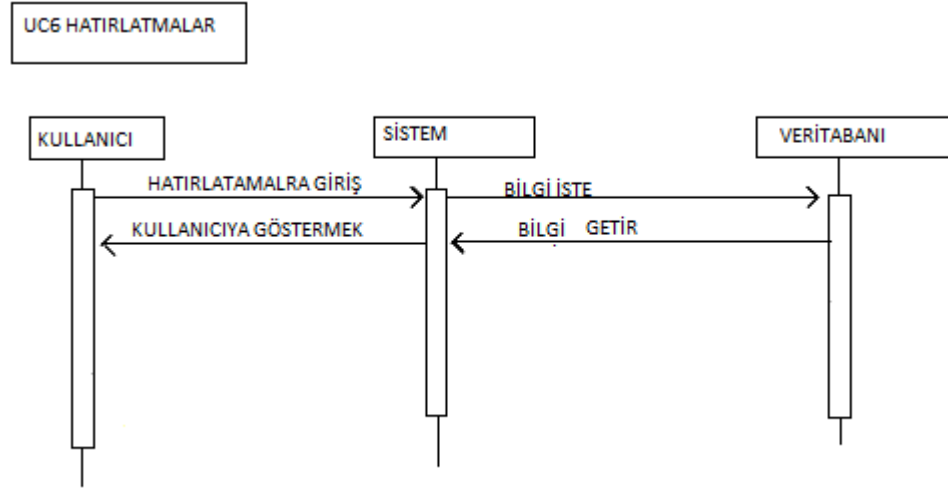
- 1.Kullanıcı uygulamaya giriş yapar
- 2.Kullanıcı uyku paneline giriş yapar
- 3.Sistem veri tabanından bilgi ister
- 4.Veritabanından bilgi gelir.
- 5.Bilgi kullanıcıya gösterilir
- 6.Hatırlatmalar gösterilir.
- Hata durumunda;
- 6.Kullanıcı düzen ekler
- 7.Düzen veritabanına kaydedilir
- 8.Bilgiler ve hatırlatmalar gösterilir.



DIYAGRAM-5 UC5 NABIZ

Diyagram-5 “UC-5 NABIZ” Use-Case 5’in dizi diyagramı gösterilmiştir. Burada kullanıcıya nabız bilgileri gösterilmektedir. Başlatıcı aktör kullanıcıdır ve aktörün hedefi nabız değerlerini ölçmektir. Katılan aktörler işlemcidir. Önkoşul, kullanıcının ana sayfadan nabız butonuna tıklaması ve bu panele giriş yapması gerekmektedir. Beklenen şart, kullanıcının nabız ölçümünü yapmak ve kullanıcıya göstermektir. Başarılı senaryo için gidilecek yol:

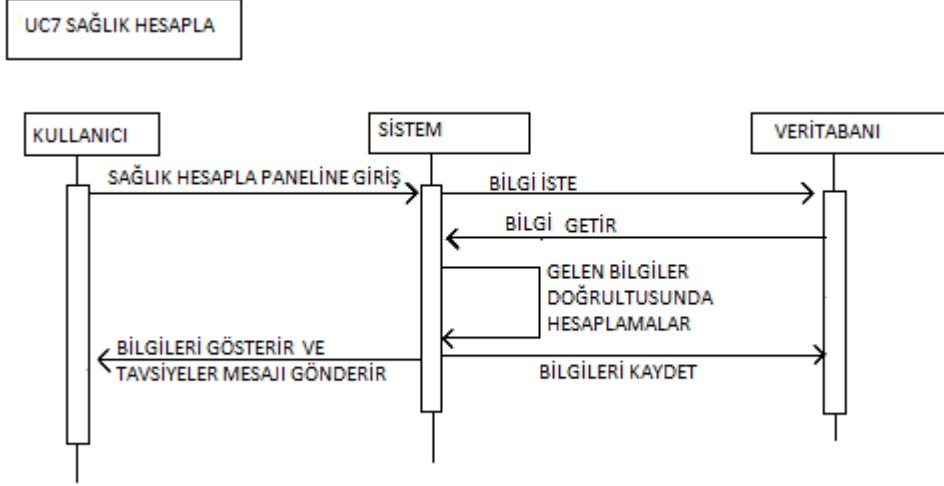
- 1.Kullanıcı uygulamayı açar
- 2.Giriş bölümünden nabız paneline giriş yapar
- 3.Sistem işlemciye sensör çalıştırma isteği gönderir
- 4.Sensör aktif olur
- 5.Sistem kullanıcıya nabız ölçmek için mesaj gönderir
- 6.Anlık kamera görüntüleri sisteme gelir
- 7.Sistem görüntüleri işlemciye iletir
- 8.İşlemci görüntü yorumlar
- 9.Görüntü yorumları sisteme iletilir
- 10.Sistem kullanıcıya iletir



DIYAGRAM-6 UC6 HATIRLATMALAR

Diyaqram-6 “UC-6 HATIRLATMALAR” Use-Case 6’nın dizi diyagramı gösterilmektedir. Kullanıcıya günlük rutin yapılması gerekenleri hatırlatmak için kullanılır. Başlatıcı aktör kullanıcıdır. Aktörün hedefi, kullanıcının günlük rutin yapması gerekenleri hatırlatılmaktır. Önkoşulu, kullanıcı uygulamaya giriş yaptıktan sonra ana sayfa üzerinde bulunan hatırlatmalar paneline giriş yapmasıdır. Beklenen şart, kullanıcıya günlük yapılması gerekenleri hatırlatılmasıdır. Başarılı bir senaryo için gidilecek yol:

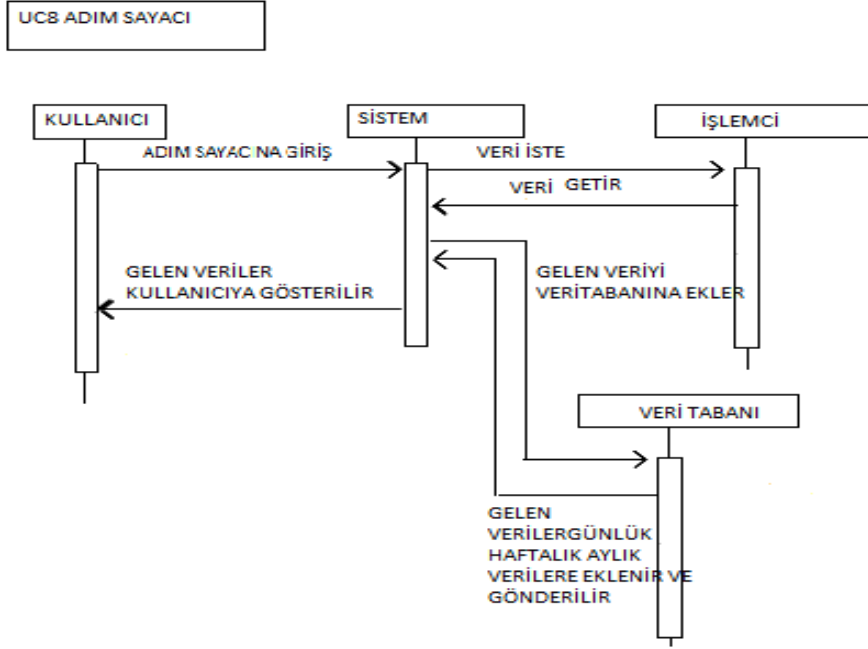
- 1.Kullanıcı uygulamaya giriş yapar.
- 2.Kullanıcı hatırlatmalar paneline giriş yapar
- 3.Sistem veri tabanından bilgi ister
- 4.Veritabanı sisteme bilgi getirir
- 5.Sistem bu bilgileri kullanıcıya gösterir.



DIYAGRAM-7 UC7 SAĞLIK HESAPLA

Diyagram-7 “UC-7 SAĞLIK HESAPLA” Use-Case 7’nin dizi diyagramı gösterilmektedir. Burada kullanıcıya Vücut kütle endeksi, yağ yüzdesi, tavsiye edilen günlük alımları hesaplar buna bağlı olarak ideal vücut ağırlığını, ideal su alma oranını ve hedef kalp hızını da hesaplayarak kullanıcıya buna uygun tavsiyelerde bulunur. Başlatıcı aktör kullanıcıdır. Aktörün hedefi, kullanıcının değerlerini girerek mevcut durumunu ve ona tavsiyelerde bulunmak için. Önkoşulu, kullanıcının bilgilerini tam olarak girmesi gerekir daha sonra da ana sayfa paneli üzerinden sağlık hesaplama butonuna tıklanması gerekmektedir. Başarılı senaryo için gidilecek yol:

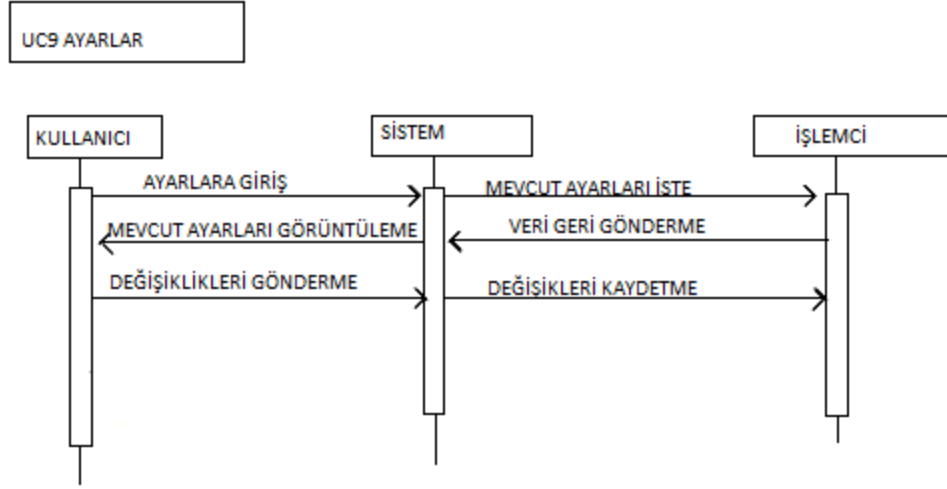
- 1.Kullanıcı uygulamaya giriş yapar.
- 2.Kullanıcı sağlık hesaplama paneline giriş yapar
- 3.Sistem veri tabanından bilgi ister
- 4.Bilgi sisteme gönderilir.
- 5.Gelen bilgiler doğrultusunda vücut kütle endeksi, yağ yüzdesi, tavsiye edilen günlük alımlar hesaplanır ve buna dayalı olarak ideal vücut ağırlığı, ideal su alma oranı ve hedef kalp hızını da hesaplayarak tavsiyeler oluşturulur.
- 6.Hesaplanan bilgiler ve oluşturulan tavsiyeler veri tabanına kaydedilir.
- 7.Hesaplanan bilgiler ve oluşturulan tavsiyeler kullanıcıya gösterilir.



DİYAGRAM-8 UC8 ADIM SAYACI

Diyagram-8 “UC-8 ADIM SAYACI” Use-Case 8’in dizi diyagramı gösterilmektedir. Burada kullanıcı günlük, haftalık, aylık olarak attığı adım sayılarını görüntülemektedir. Başlatıcı aktör, kullanıcıdır. Aktörün hedefi, kullanıcının hareket aktivitesine bağlı olarak elde ettiği değerleri görmesidir. Katılan aktörler, işlemci ve veri tabanıdır. Önkoşulu, kullanıcının uygulamayı açması sonra giriş sayfasından (ana sayfadan) adım sayacı paneline giriş yapması gerekmektedir. Beklenen şart, işlemci kullanıcının günlük attığı adıma göre gidilen mesafeyi, yürüme süresini ve yakılan kaloriyi hesaplar ve hesaplanan bu değerleri kullanıcıya görüntülenmesi sağlanır. Başarılı senaryo için gidilecek yol:

- 1.Kullanıcı uygulamaya giriş yapar.
- 2.Kullanıcı adım sayacı paneline giriş yapar
- 3.Sistem işlemciden veri (Gps üzerinden atılan adımlar hesaplanmış hali) ister
- 4.veri sisteme gelir
- 5.Gelen veri veri tabanına eklenir
- 6.Gelen veriler günlük, haftalık, aylık olarak diğer verilere eklenir.
- 7.Eklenmiş veriler sisteme iletilir.
- 8.Veriler günlük, haftalık, aylık olarak kullanıcıya gösterilir.



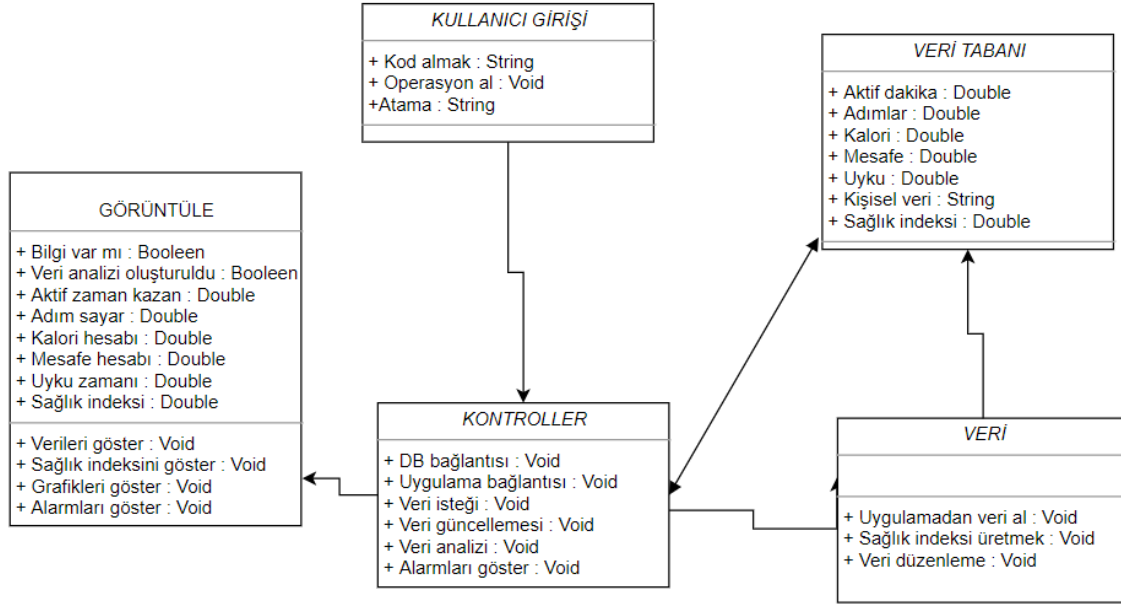
DİYAGRAM-9 UC9 AYARLAR

Diyagram-9 “UC-9 AYARLAR” Use-Case 9’un dizi diyagramı gösterilmektedir. Kullanıcı uygulama ayarlarını kendine özgü özelleştirmesi için kullanılır. Başlatıcı aktör, kullanıcıdır. Aktörün hedefi, kullanıcı uygulamanın ayarlarını özelleştirebilmesidir. Katılan aktör, işlemcidir. Önkoşul, kullanıcının uygulamayı açması daha sonra ana sayfa üzerinde bulunan ayarları butonuna tıklayıp bu panele giriş yapması gerekmektedir. Başarılı senaryo için gidilecek yol:

- 1.Kullanıcı uygulamaya giriş yapar.
- 2.Kullanıcı ayarlar paneline giriş yapar
- 3.Sistem işlemciden mevcut ayarları ister
- 4.Veriler sisteme gönderilir
- 5.Sistem kullanıcıya mevcut ayarları görüntülenmesini sağlar
- 6.Kullanıcı değişiklikler yapar
- 7.Kullanıcı yaptığı değişiklikleri sisteme gönderir
- 8.Sistem değişiklikleri kaydetmesi için işlemciye gönderir.

2-)SINIF DİYAGRAMI VE ARAYÜZ ÖZELLİKLERİ

A. Sınıf Diyagramı



B. Veri Türleri ve İşlem İşaretleri

1. Ekran: Kullanıcı ara yüzüne dayanır.

- Uygulama Bilgi: Veriler kullanıcı tarafından girildiğinde bu verilere karşılık gelen değişkenimiz.
- Veri Analizi Üretildi mi: Kullanıcınızın sağlığı hakkında tüm bilgiler girilmiş olup, bu bilgiler nazarında sonuçları uygulanmaktadır. Girilmiş olan verileri temizlemek, dönüştürme ve modelleme işlemleri yapılmaktadır.
- Aktif Dakika: Kullanıcımızın, üretmiş olduğumuz uygulamamızda geçirmiş olduğu aktif süredir.
- Adımlar: Kullanıcımızın sağlığı açısından belirlemiş olduğu günlük adım sayısına karşılık gelen değişkenimizdir.
- Kalori Al: Sağlık dostu cihazımızda kaydedilen, kullanıcımızın gün içerisinde almış olduğu kalorisine denk gelen değişkenimiz.
- Mesafe: Kullanıcımızın adım sayılarımızda belirlediği adım sayısına göre mesafenin ölçülüp girilmesi. Mesafenin girilmesi.

- Uyku: Kullanıcıma göre en sağlıklı uyku düzeni ve saatinin belirlenip girilmesi işlemi.
- Sağlık Dostu Verilerini Göster: Ana sayfa, günlük uyku, hedef, nabız, hatırlatmalar, sağlık hesaplama, adım sayacı ve ayarlar.
- Sağlık İndeksi: Yapmış olduğumuz uygulamamızdaki verileri ve sağlık durumunu gösterir. Kullanıcımızın tüm verileri her gün değişmektedir. Bir önceki günü baz alarak bir sonraki günde olan değişimi tahmin edebilmekteyiz. Bir bütün olarak düşünecek olursak bu döngüyü kişinin sağlık indeksi hakkında ilerleyen zamanlarda yorum yapabiliriz.
- Grafik Gösterimi: Kullanıcımızın verilerine, sağlık indeksine bakarak sağlığı hakkında bir grafik oluşturulmaktadır.
- Showline Grafikleri: Uygulamamızdan alınan kullanıcı bilgilerini çizgi grafiklerine her şey dâhil aktif gösterme. Kullanıcımızın sağlık durumunu daha iyi gözlemlemek için bu grafikleri görüntüleyebiliriz. Bu grafiğimiz bize kullanıcımız hakkında kapsamlı bir bilgi akışı sunar.
- Tavsiye Et: Kullanıcımızın sağlık durumuna göre onun sağlığı hakkında uygulamamızda yer alan kategorilerimizden hepsini ayrı ayrı baz alarak en iyi durumunu bulup tavsiyelerde bulunma işlemidir.
- Sağlık İndeksi Göstergesini Göster: Kullanıcımızın her gün yapmış olduğu sağlık indeksi veri sonuçlarını göstermesi.

2. Kullanıcı Girişi

Bizim kullanıcımızın sağlık durumu hakkında kapsamlı bir bilgi sahibi olmak için toplanan veriler bütünü oluşturur ekranımız. Amacı kullanıcımızı tanımak.

- Boş Kod Dizisi: Kullanıcı girişi boş olan ekran. (String)
- Operasyon: Kullanıcımızın kendi sayfasında yapmış olduğu işlemleri gösterir. Mesela vücuda alınan su miktarının girilmesi. (Void)
- Durum Dizgisi Al: Kullanıcının erişmek istediği bilgilerini sorgulaması. (String)

3. Denetleyici

Uygulamamızda kullanıcımızın takip ettiğimiz sağlık durumu aktiviteleri birbirine bağlantılıdır. Mesela adım sayar ile birlikte kalori miktarındaki düşüş birbiri ile bağlantılı şekilde çalışır. Veya yürüyüş esnasındaki nabız atımı ile normaldeki nabız atımı farklıdır. BU yüzden denetleyicimiz bu işlemlerden, bağlantılarımızdan sorumludur.

- Veri Tabanı Bağlantısı: Sağlık dostu verileri için, veri tabanı bağlantısı kurar. (Void)
- Sağlık Dostu Bağlantısı: Belirli bir kullanıcı için uygulamanın veri tabanına bağlantı kurar. (Void)

- Veri İsteği: Sorguları veri tabanına göndermek ve almak için genel bir yöntem görevi görür. Veri tabanımızın bize gelen yanıtı bu işlev ekran tarafından çağrılacaktır. (Void)
- Veri Güncelleme: Verilerimizin güncellenmesi işlevi kullanıcılar tarafından yapılmaktadır. Girmiş olduğu eski verilere silinerek yerine yeni veriler yerleştirilir. Ve bir sonraki aşamaya hazır olur. (Void)
- Veri Analizi: Verileri analiz etmek ve sağlıklı birey oluşturmak için kullanılan VHO(World Health Organization) algoritmasını kullanmak için tasarlanmıştır. Veri tabanından veriler kullanıcı istediğinde çağırılır. (Void)
- Tavsiye Alın: Kullanıcımız en sağlıklı nasıl olunur? Sorusunun cevabının bulunması için tasarlanmıştır. Kullanıcının sağlık endeksi hesaplanarak ve en kötü durum baz alarak tavsiyeler üretilir. Kullanıcımızın aktif olarak uygulamamıza girdiği zamanları seçerek o zamanı baz alarak tavsiyede bulunuruz. (Void)

4. Veri Tabanı

Bu sınır verileri depolamak için oluşturulmuştur. Saklanması gereken bazı değişkenleri tanımlar.

- Aktif Dakika: Kullanıcımızın bizim uygulamamızda bitirmiş olduğu sürece karşılık gelir. (Double)
- Adımlar: Kullanıcımızın günlük atmış olduğu adımın kayıtlı olduğu kısım. (Double)
- Kalori: Sağlık dostu cihazımıza kaydedilen, kullanıcımıza ait kalori hesabına karşılık gelen değişken. (Double)
- Mesafe: Atmış olduğu adıma göre kayıt edilen mesafe bilgisi.(Double)
- Uyku: Kullanıcımızın günlük uyku düzenine göre cihazımıza kaydedilen uyku süreci. (Double)
- Sağlık Endeksi: Kullanıcımızın boy, kilo, uyku, kalorisine göre oluşturulan sağlık endeksi bilgileridir. (Double)

5. Veri İşlemcisi










Uygulamamızdan alınan verileri analiz etmek ve verileri veri tabanında saklamak için oluşturulmuştur.

Fonksiyonlar:

- Uygulamalardan Bilgi Al: Denetleyicisi uygulamayı bitirdikten sonra yeni veri almak için bu yöntemi çağırır. (Void)
- Sağlık Endeksi Oluştur: Sağlık endeksi oluşturmak için kullanıcımızın sağlık durumu ile alakalı tüm veriler toplanır ve bir hesaplama gerçekleşir. Bu hesaba dayanan sistemdir. (Void)

- Veri Yazma isteği: Kullanıcımızı sağlık endeksine verileri saklamasını istemek için inşa edilmiştir. Ve o kullanıcı için izin oluşturduktan sonra çağrılır.

C. İZLENEBİLİRLİK MATRİSİ

ETKİ ALANI KAVRAMI	SINIFLAR				
	GÖRÜNTÜLE	KULLANICI GİRİŞİ	KONTROLÖR	VERİ TABANI	VERİ ANALİZİ
ARAYÜZ					
KONTROLÖR					
DENETLEYİCİSİ					
KONNEKTÖR					
BİLGİ DEPOLAMA					
İLETİŞİMCİ					
METİN OKUYUCU					
SAĞLIK ANALİZİ					

1.1 İZLENEBİLİRLİK MATRİSİ


D. NESNE KISITLAMA DİLİ (OCL) SÖZLEŞMELERİ

1.Ekran

Görüntüle	
Değişmezler	Kullanıcının eklediği kod kimliği
Ön koşul	Veriler programımızdan alınır ve oluşturulur.
Hedef Şart	Uygulama sayfalarını kullanıcıya göster

1.2 OCL SÖZLEŞMELERİNİ GÖSTER

2. Kontroller

Kontrolör	
Değişmezler	Programdan veri almak, sağlık dizini oluşturmak gibi eylemler.
Ön koşul	Diğer sınıflardan talep alınır.
Hedef Şart	Başka bir sınıfa talep veya sonuç döndürme  (Ctrl) ▾

1.3 KONTROLLER OCL SÖZLEŞMESİ

3.Kullanıcı Girişi

Kullanıcı Girişi	
Değişmezler	Kullanıcının yaptığı işlemler veya kullanıcının eklediği metin
Ön koşul	Kullanıcının bir parmak izi
Hedef Şart	Kullanıcının aramak istediği işlev anlamına gelen bir değer döndürün

1.4 KULLANICI GİRİŞİ OCL SÖZLEŞMESİ

4.Veritabanı

Veritabanı	
Değişmezler	Kullanıcının program ve sağlık dizininden verileri ve ayrıca sigara içme ve egzersiz koşulu gibi sağlık verileri.
Ön koşul	Programa erişim ve verileri güncelleyin
Hedef Şart	Program verilerini oluşturdu ve verileri güncelledi

1.5 VERİ TABANI OCL SÖZLEŞMESİ

5. Veri İşlemcisi

Veri işlemcisi	
Değişmezler	Sağlık indeksinin hesaplanması ve farklı sağlık indeksinin hesaplanması
Ön koşul	Kullanıcıdan alınan veriler
Hedef Şart	Sağlık endeksi ve veri tabanında oluşturulan diğer analizler

1.6 VERİ İŞLENCİSİ OCL SÖZLEŞMESİ

3-) SİSTEM MİMARİSİ VE SİSTEM TASARIMI

A. MİMARİ STİLLER

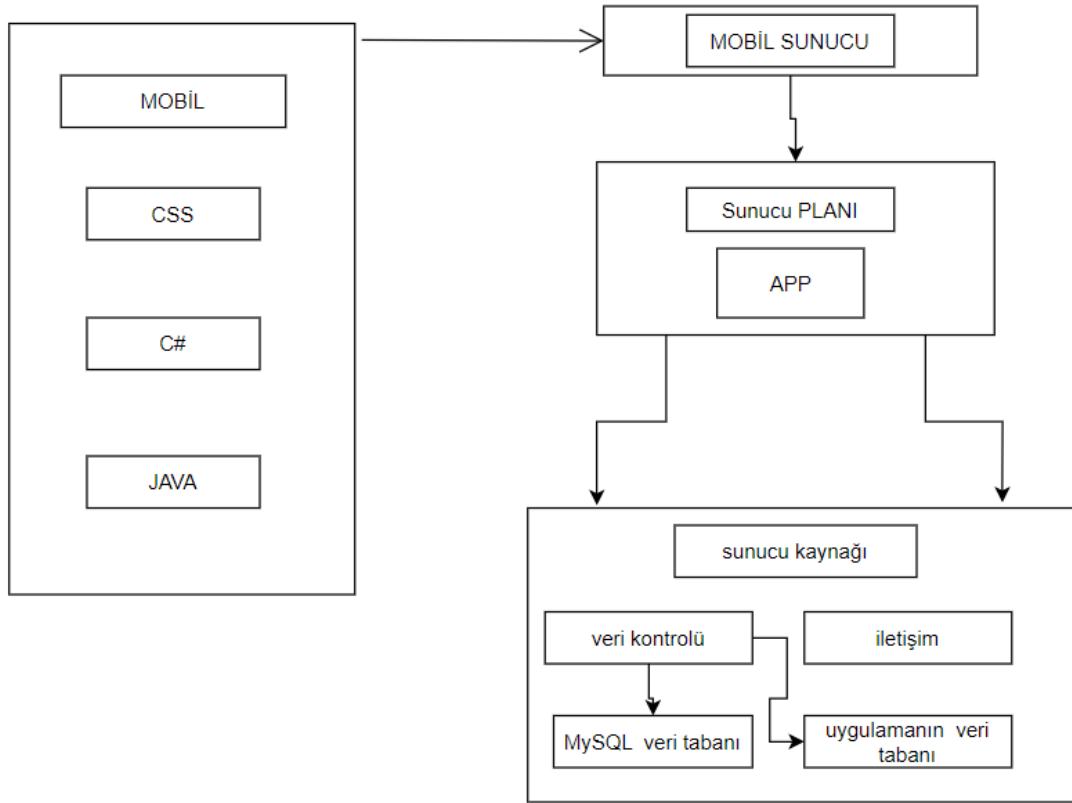
Uygulamamızda bir istemci ve sunucu mimari modeli vardır. İstemci ve sunucu modelimizde, dataları tutmak için bir adet merkezi sunucu vardır. Uygulamamızı kullanan kullanıcıların her biri ayrı şekilde aynı anda bir uygulamaya girebilir. Sunucu veri depolama ve veri hesaplamalarının çoğunu yapabilecektir. Toplanan bütün verilerin depolanması için bir veri tabanı kullanılacak. Programımızda girmesi istenen verilerin hesaplanması ve sonuca yönelik işlemleri yapabilmek için bir algoritma kullanılacak.

İstemci sunucu ile zahmetsizce iletişim kurar ve sunucumuzda yer alan verilerimizi görüntüleme işlemi yapar. Uygulamamız ve kullanıcımız arasındaki verimli iletişim XML ile yapılır. İstemcimiz verilere erişmek için bir istekte bulunur ve bu istek sonrasında sunucumuz tarafından onaylanır.

İstemcimizin isteği doğrultusunda sunucumuzdaki veriler kullanıcımıza sunulur. Bu şekilde bir mimari kullanınca, bize kullanıcı ve uygulamamız arasındaki etkili iletişimi arttırdığı ve verimli geçtiğini söyleyebiliriz.

Mimarimiz oldukça hızlı ve verimli çalışır. Az depolama ve hızlı bir program sunar. Herhangi bir donma veya uygulamayı kaldıramama gibi bir problem teşkil etmez. Kullanıcımızın girmiş olduğu verileri doğru şekilde hesaplar ve yanılma payı olmadan doğru sonuca bizi götürür. Hesaplama sunucumuz oldukça güvenilirdir.

B. Alt Sistemlerin Tanımlanması



1.7 UML DİYAGRAMI

Bizim uygulamamızda iki sistemimiz vardır. Bir tarafı mobil iken diğer kısım mobil sunucudur. Mobil tarafımız bize sunucu tarafımızın iskelet yapısını oluşturur. Mobil tarafımız mantık çerçevesinde analiz eder ve daha sonra sunucu tarafına iletir.

Sağlık dostu veri tabanı ve MySQL veri tabanından veri alır.

C. Alt Sistemlerin Donanıma İşlemi

Mobil paketimiz istemcimizin uygulamasına ayrılır ve kullanıcı buna erişmek için kullanır. Mobil sunucusu ve mobil kullanıcı taleplerini en iyi şekilde analiz etmek ve sonuca bağlamak için kullanırlar.

D. Kalıcı Veri Depolama

Verilerimi saklamak için MySQL veri tabanı seçilir. MySQL veri tabanı bize geniş platformda bilgiler sunar.

Programımızda bulunan her seçenek için etkin durumdadır. Bize depolama olanağını yüksek seviyede sunar ve tablolarımız için oldukça uygundur. Bazı verilerimiz kullanıcılarımızın ihtiyaçları doğrultusunda çıkarken bazı verimiz kalıcı depolamaya ihtiyaç duyar.

Yaptığımız programın veri tabanı kullanıcılarımıza kalıcı depola işlemi ile geçmişteki verilerini göstererek bugün ile değişimlerinin kıyaslanmasına yardımcı olur. Yapılan kalıcı depolama işlemi uygulamamızda buluna seçeneklerimizin büyük kısmını kapsamaktadır.



1.8 VERİ TABANI ŞEMASI

E. Ağ Protokolleri

TCP (İNTERNET AĞ MİMARİSİ) işletim sisteminin arka planda yürüttüğü programlar olarak adlandırılırlar.

SMTP (SİMPLE MAİN TRANSPOT PROTOKOLÜ) kullanıcı bilgilerini denetler.

SFTP (FILE TRANSFER PROTOCOL- DOSYA AKTARIM PROTOKOLÜ) Bir telefonda diğer telefona veri aktarımını sağlar.

F. GLOBAL KONTROL AKIŞI

Kullanıcı uygulamamıza giriş yaparak, yazarak, tıklayarak, hareket ettirerek veri girebilir. Kullanıcı uygulamamızı rahatlıkla kullanabilir. Kullanıcılarımız, giriş ekranımıza girip bilgilerini doldurduktan sonra kendilerine ait tüm verileri görebilir ve değişim yapabilir.

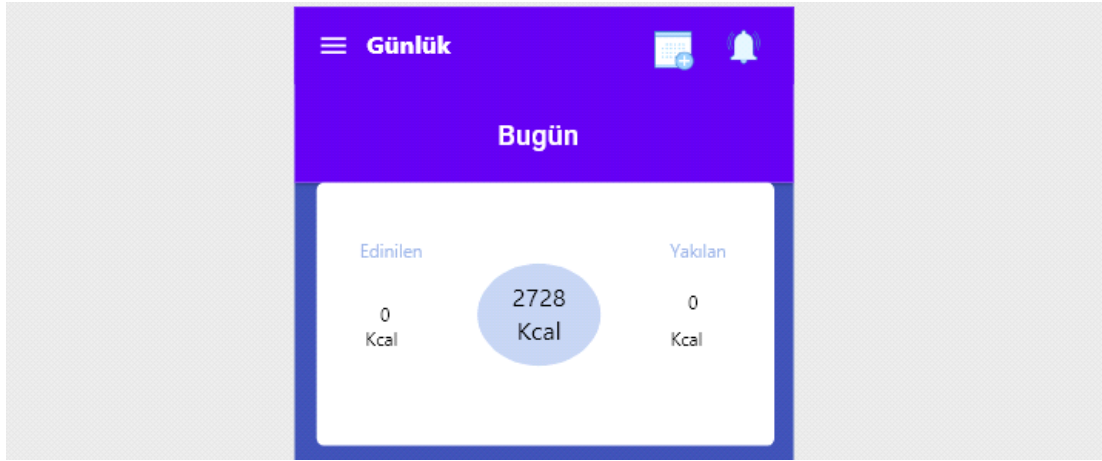
D. DONANIM GEREKSİNİMLERİ

Uygulamamızın işletim sistemi IOS ve ANDROID olabilir. Depolama alanı 15-20 MB'tan büyük olmalıdır. 120 MB'tan küçük olmalıdır.

Veri tabanı ve MySQL veri tabanı sistemimizin veri tabanlarıdır.

4-)Algoritmalar ve Veri Yapıları

A-)ALGORİTMALAR

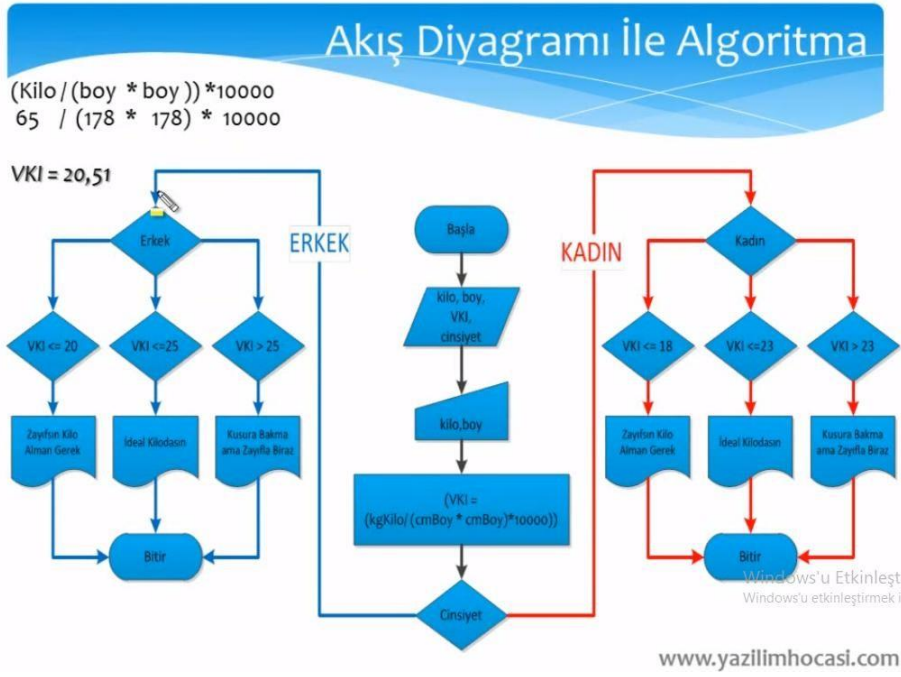


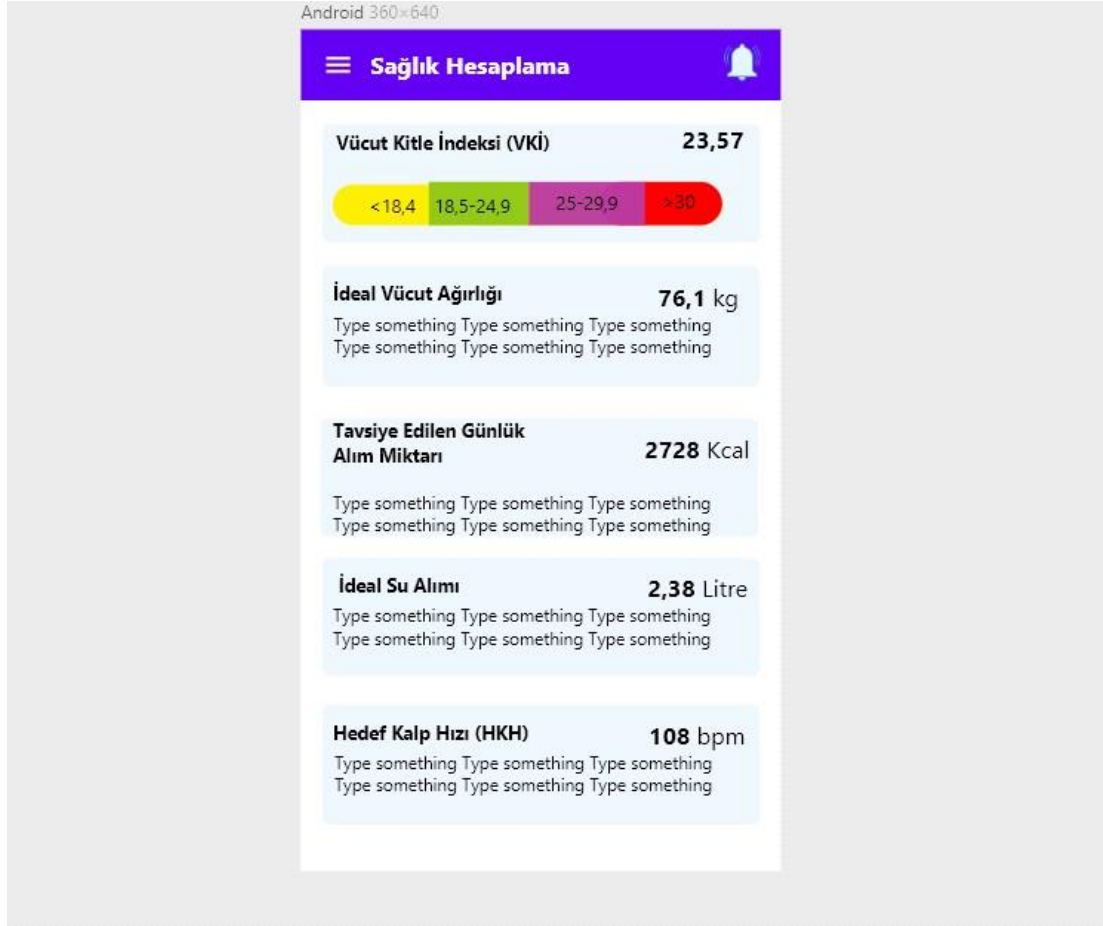
Net kalori hesaplama :

Günlük edinilen kaloriden, yakılan kaloringin çıkarılması ile oluşur.

$$\text{Edinilen} - \text{Yakılan} = \text{Net Kcal}$$

Vücut Kitle İndeksi (VKİ) :





BoyKiloEndeksi veya diğer adıyla Vücut Kitle Endeksi, vücudunuzdaki tahmini yağ oranını gösteren bir değerdir.

BoyKiloEndeksi hesaplamasında boy-kilo oranı kullanılır. Dolayısıyla BoyKiloEndeksi değeri, boyunuza göre ideal kilonuzun kaç olması gerektiğini gösterir. Vücut ağırlığınız, boyunuzun karesine bölünerek boy kilo cetveli üzerinde ideal kilonuz hesaplanır.

Algoritmanın örnek kod gösterimi :

```
boy = float(input("Boy:"))
```

```
kilo = int(input("Kilo:")) boy
```

```
= boy/100;
```

```
endeks = kilo/(boy^2)
```

```
if endeks<18.5:
```

```

        print("\n Zayıf")
elif endeks > 18.5 and endeks <= 25 :
        print("\n Normal")
elif endeks > 25 and endeks <= 30:
        print("\n Kilolu")
elif endeks > 30:
        print("\n Obez")

```

İdeal Vücut Ağırlığı :

İdeal kilo sağlıklı bir vücudun vücut yüzeyine (boy) göre kilo oranı hesaplanarak bulunur.

Algoritmanın örnek kod gösterimi :

```

int kilo,boy; Console.WriteLine("Kilonuzu
giriniz = ");
kilo = Convert.ToInt32( Console.ReadLine() );
Console.WriteLine("Boyunuzu giriniz = "); boy =
Convert.ToInt32( Console.ReadLine() ); boy = boy %
100;
if( boy - kilo > 11){
Console.WriteLine("Kilo almanız gerek");} if(
boy - kilo == 11 ){
Console.WriteLine("Kilonuz ideal"); }
else{
Console.WriteLine("Kilo vermeniz gerek."); }

```

- 1-)Başla
- 2-)kilo,boy değişkenlerini al
- 3-)kilo,boy değerini gir
- 4-)boy = boy % 100
- 5-)Eğer boy – kilo > ise 8'e git,değilse devam et
- 6-)Eğer boy – kilo = 11 ise 9'a git,değilse devam et
- 7-)Eğer boy – kilo < 11 ise 10'a git,değilse devam et
- 8-)Yazdır kilo alman gerek ve 11'e git
- 9-)Yazdır ideal kilo 11'e git
- 10-)Yazdır kilo vermengerek
- 11-)Bitir

Günlük kalori ihtiyacı :

"Bazal metabolizma", "egzersizde" ve "diyetle alınan besinlerin sindiriminde" harcanan enerji olmak üzere üç ana bileşenden oluşur. Bunların arasındaki en büyük belirleyici bazal metabolizma hızıdır. BMH, dinlenirken veya uyurken yani hiçbir aktivite gösterilmediğinde harcanan enerji miktarını göstermekte ve günlük harcanan kalenin yüzde 60 ile 80'ini oluşturmaktadır.

Fiziksel aktivitede enerji tüketiminde, yani kilo verme veya alma üzerindeki önemli faktörlerden biridir. Fiziksel aktivitenin çoğaltılması durumunda metabolizma hızı, dolayısıyla günlük kalori ihtiyacı da arttırılmış olur.

Bazal metabolizma hızı nedir?

Kilo vermek isteyen bir kişinin diyetle ne kadar zayıflayacağını belirleyen en önemli faktördür. Örneğin; ağızdan alınan şekerin karaciğerde yağa dönüştürülüp, cilt altı yağ dokusunda depolanmasında olduğu gibi vücuttaki maddelerin birbirine dönüşmesini, biyolojik olayların devamını sağlamak için ne kadar enerji yakılması gerektiğini göstermektedir. Genler, cinsiyet, yaş, vücut ağırlığı, vücut ısı ile egzersiz gibi pek çok faktörden etkilenmektedir.

Kadınlarda bazal metabolik hızının algoritması:

$$655 + 9.6 \times (\text{kg cinsinden ağırlık}) + 1.8 \times (\text{cm cinsinden boyu}) - 4.7 \times (\text{yaş}) \text{ Erkeklerde}$$

bazal metabolik hızının algoritması:

$$66 + 13.7 \times (\text{kg cinsinden ağırlık}) + 5 \times (\text{cm cinsinden boyu}) - 6.8 \times (\text{yaş})$$

İdeal su alımı :

Kilonuza göre tüketmeniz gereken su miktarını öğrenmek için vücut ağırlığınızın değerini libre olarak hesaplamanız ve ikiye bölmeniz gerekir. Bu matematiksel işlemin sonucu, vücudunuzun günlük ihtiyaç duyduğu suyun ons cinsinden miktarını verir.

Örneğin: 116 lbs (52 kg) olan birisinin kilosunu ikiye bölelim. Sonuç 58 ons (1,7lt), bu da 5 bardak suya denk geliyor.

Hatırlanması gereken formül: Libre cinsinden vücut ağırlığı/2 = Ons cinsinden günlük su tüketimi.

Hesaplama: 1 lbs yaklaşık 0,45 kg; 1 oz yaklaşık 30 ml olarak hesaplanabilir. Eğer birim dönüştürmekle uğraşmak istemezseniz, yaklaşık bir değer bulmak için kilonuzu 30'a bölebilirsiniz. Yukarıdaki örneği tekrarlırsak:

52kg/30=1,73lt.

9kilo:0.25 litre

18kilo:0.5 litre

27 kilo: 0.75 litre

36 kilo: 1 litre

45 kilo: 1.25 litre

54 kilo: 1.5 litre

63 kilo: 1.75 litre

72 kilo: 2 litre

81 kilo: 2.25 litre

90 kilo: 2.5 litre

99 kilo: 2.75 litre

108 kilo: 3 litre

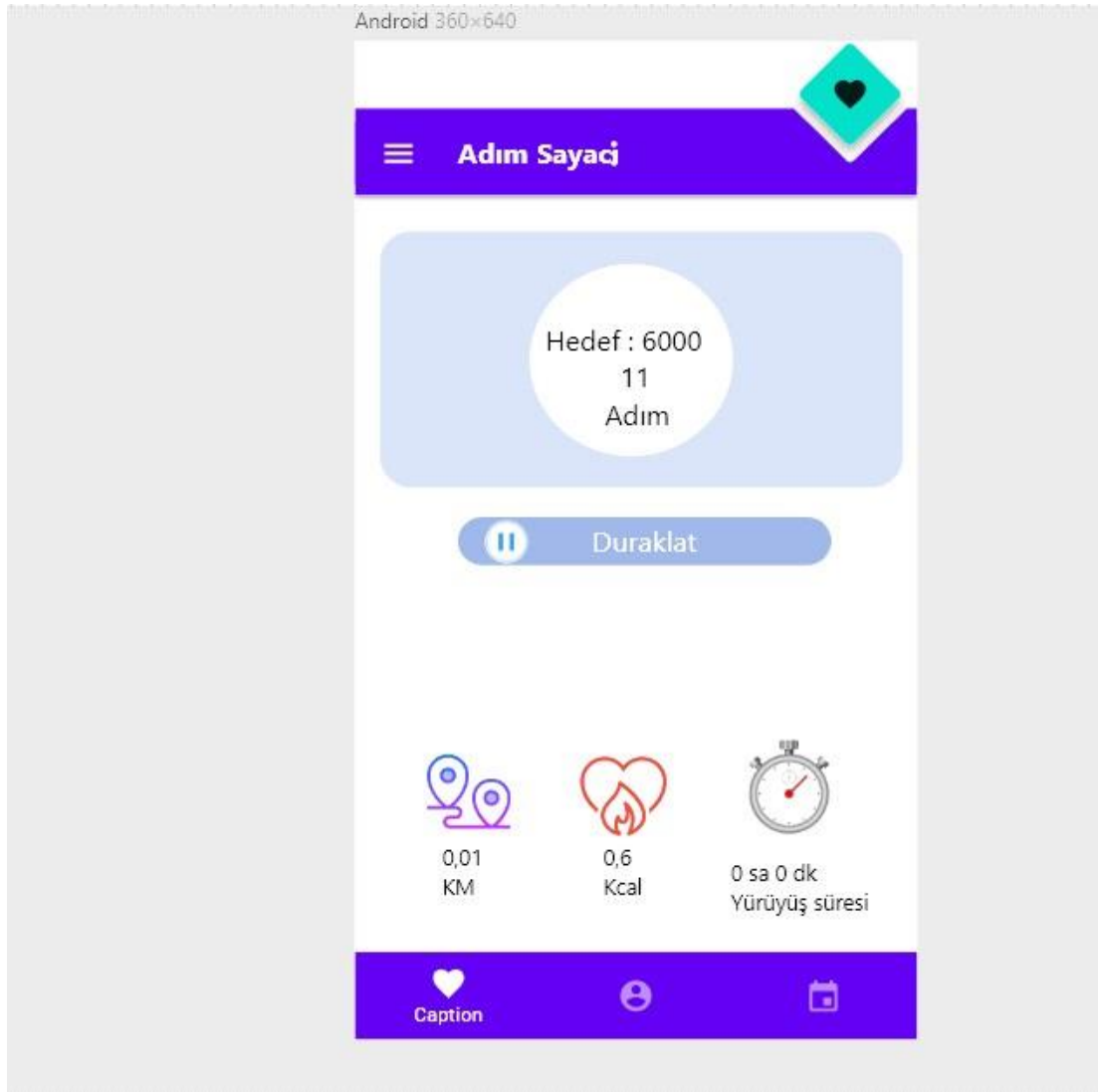
117 kilo: 3.25 litre

Hedef kalp hızı (HKH) :

Amaç özellikle akciğerler ve kalbinizin sağlığı ise egzersizin ağırlığı hakkında kalp hızınız oldukça yararlı bilgi verir: kalbinizin ulaşabileceği en yüksek hız maksimum kalp hızıdır. Bu hız = $220 - \text{yaşınız}$ dır. Yani 45 yaşında iseniz maksimum kalp hızınız $220 - 45 = 175$ olacaktır.

Kalp ve akciğer sağlığı açısından en iyi yarar, egzersiz sırasında kalbinizin hızı maksimum kalp hızının %50'si ile %75'i arasında ise alınır. Bu aralıktaki kalp hızına hedef kalp hızı denir. Hedef

nabız = $\text{Dinlenik nabız} + \text{yaş} (220 - \text{dinlenik nabız} - \text{yaş})$



Adım uzunluğu :

Boy uzunluğunun yüzde 42'si

$$\text{Adım_uzunlugu} = (\text{boy_uzunlugu} * 42) / 100$$

$$\text{Atılan_adım} = (\text{gidilen_km} / 100000) / \text{adım_uzunlugu} \text{ 1 KM}$$

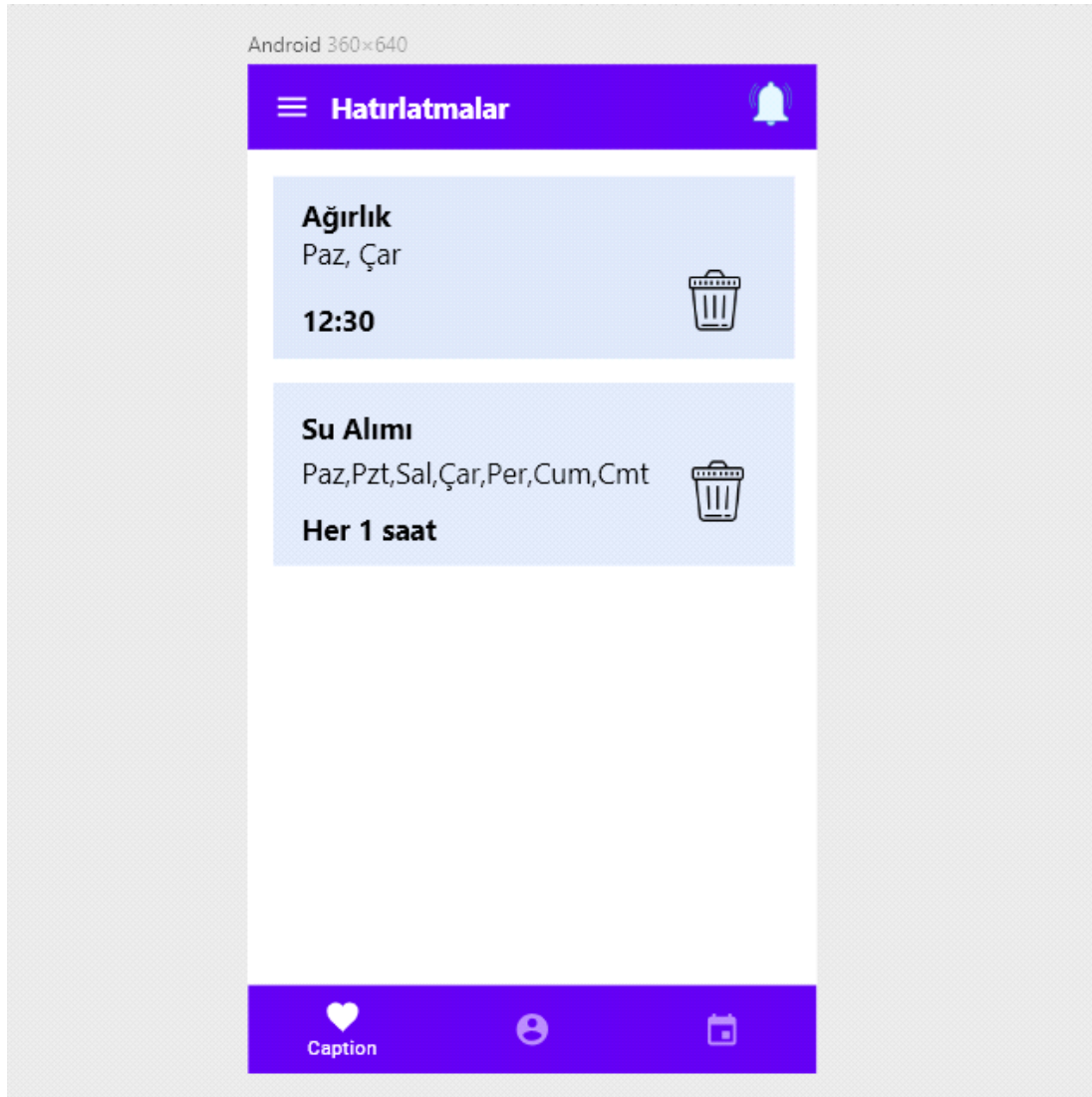
= 1000 metre

1metre=1000cm

Adıma göre yakılan kalori:

Bir adım 0.05 kalori

$$\text{Yakılan_kalori} = \text{Atılan_adım} * 0.05 ;$$



Günlük kalori ihtiyacı - Net Kcal = Günlük Alınmamış kalori

ToplamAlınmamışkalori=GünlükAlınmamışkalori+ToplamAlınmamışkalori if(7000

<= Toplam Alınmamış kalori)

{

Güncel Kilo = Kilo- 1 ;

}

Toplam Alınmamış kalori = 0 ;

B. VERİ YAPILARI

Verilerimiz çoğunlukla uygulamamızdan toplanmaktadır .Uygulamamız kullanıcılarımıza bazı veriler girmelerini ister.

Sağlık indeksini hesaplamak için ihtiyaç duyduğumuz tüm veri türleri yukarıda verdiğimiz ekran görüntüleri ve metinlerle gösterilmektedir.

5-) Kullanıcı Arayüz Tasarımı ve Uygulaması

1. Giriş Ekranı

Android 360x569

Artboard 287x424

Kullanıcı Oluştur

İsim

Doğum Tarihi
25-03-2005

Cinsiyet
Erkek

Yaş
15

Ağırlık

Yükseklik

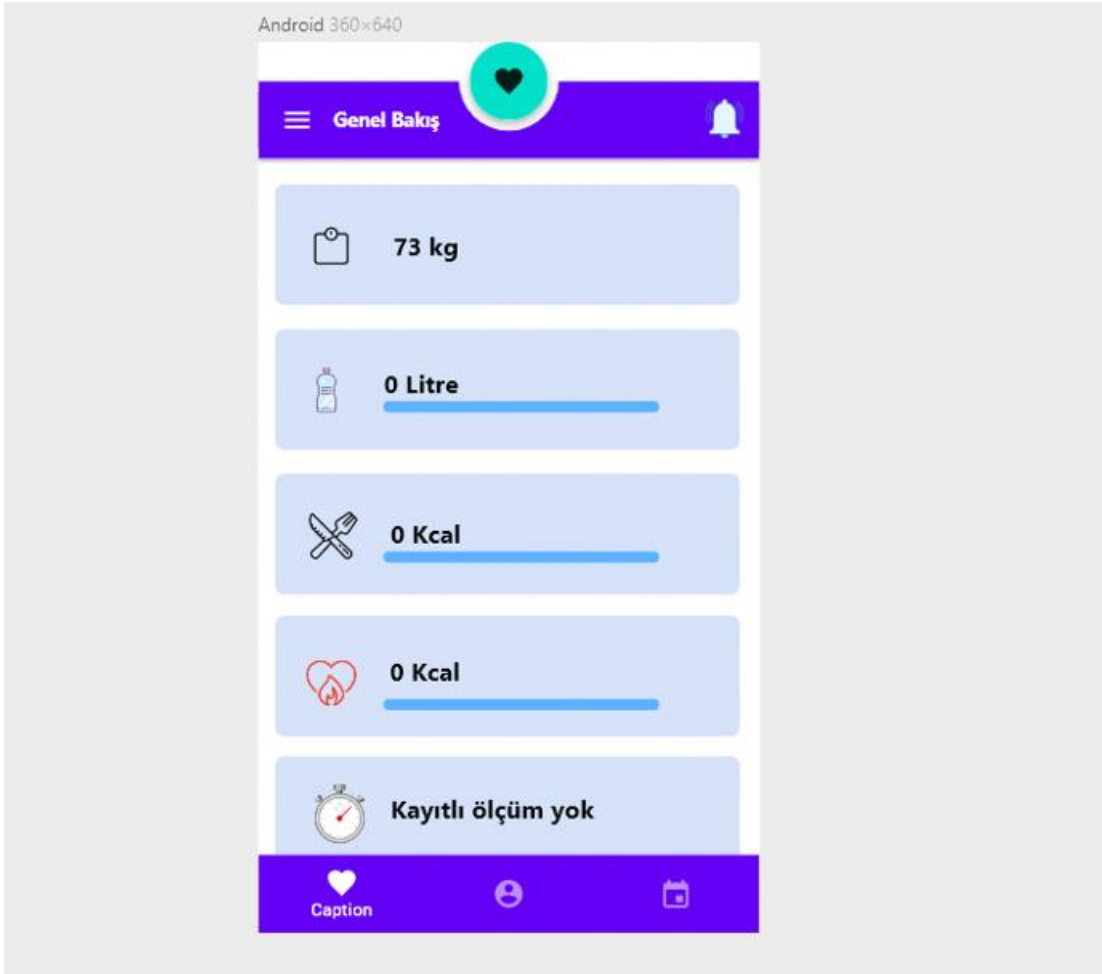
Navigation Bar: Heart, Person, Calendar icons

Caption

2.1. Giriş Ekranı

İlk olarak karşımıza bir giriş ekranı gelmekte ve bizim verilerimizi girmemiz istenmektedir. Bu verileri biden istemesindeki amaç vücudumuzu anali etmek istemesidir. Doldurmanız için İsim, doğum tarihi, ağırlık (kg) ve yükseklik (boy) kısımları bulunmaktadır.

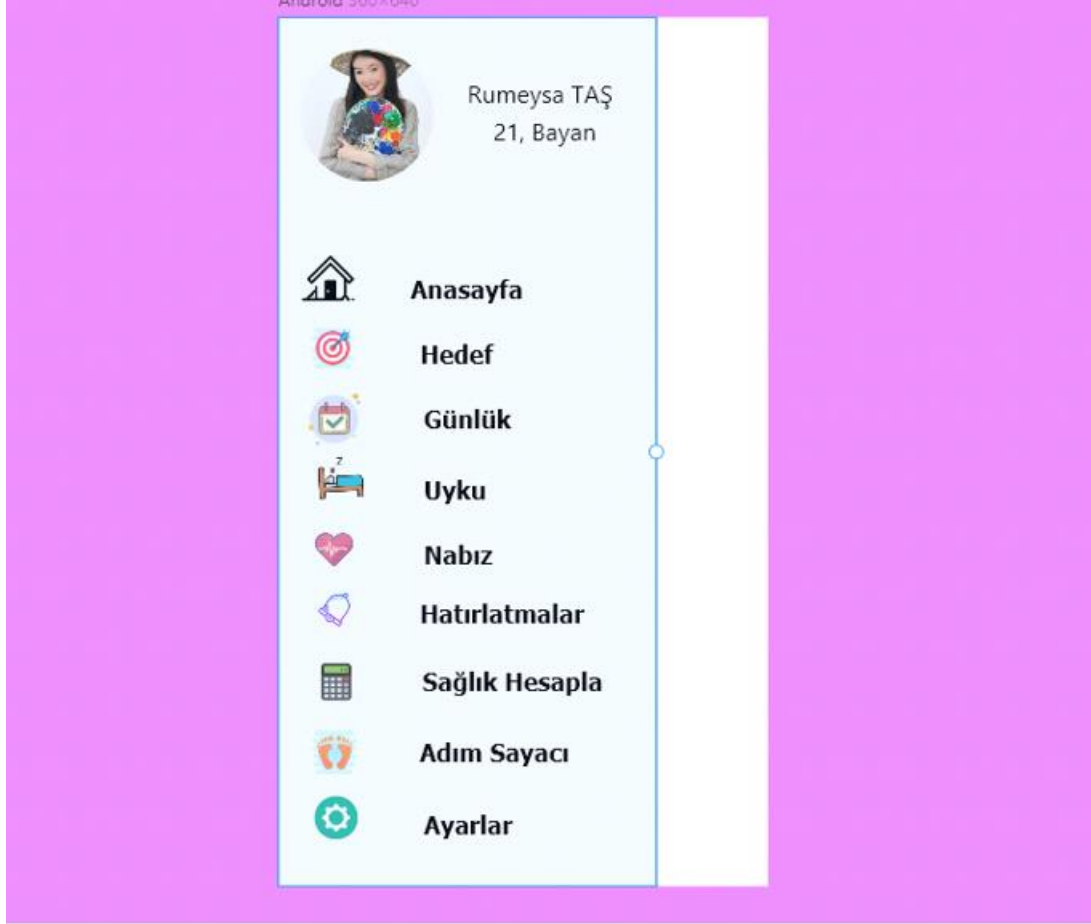
2. Genel Bakış



2.2. Genel Bakış

Bu ekranda kullanıcımızın vücudu hakkında bilgiler yer almakta ve programın ölçeceği değerler bulunmaktadır.

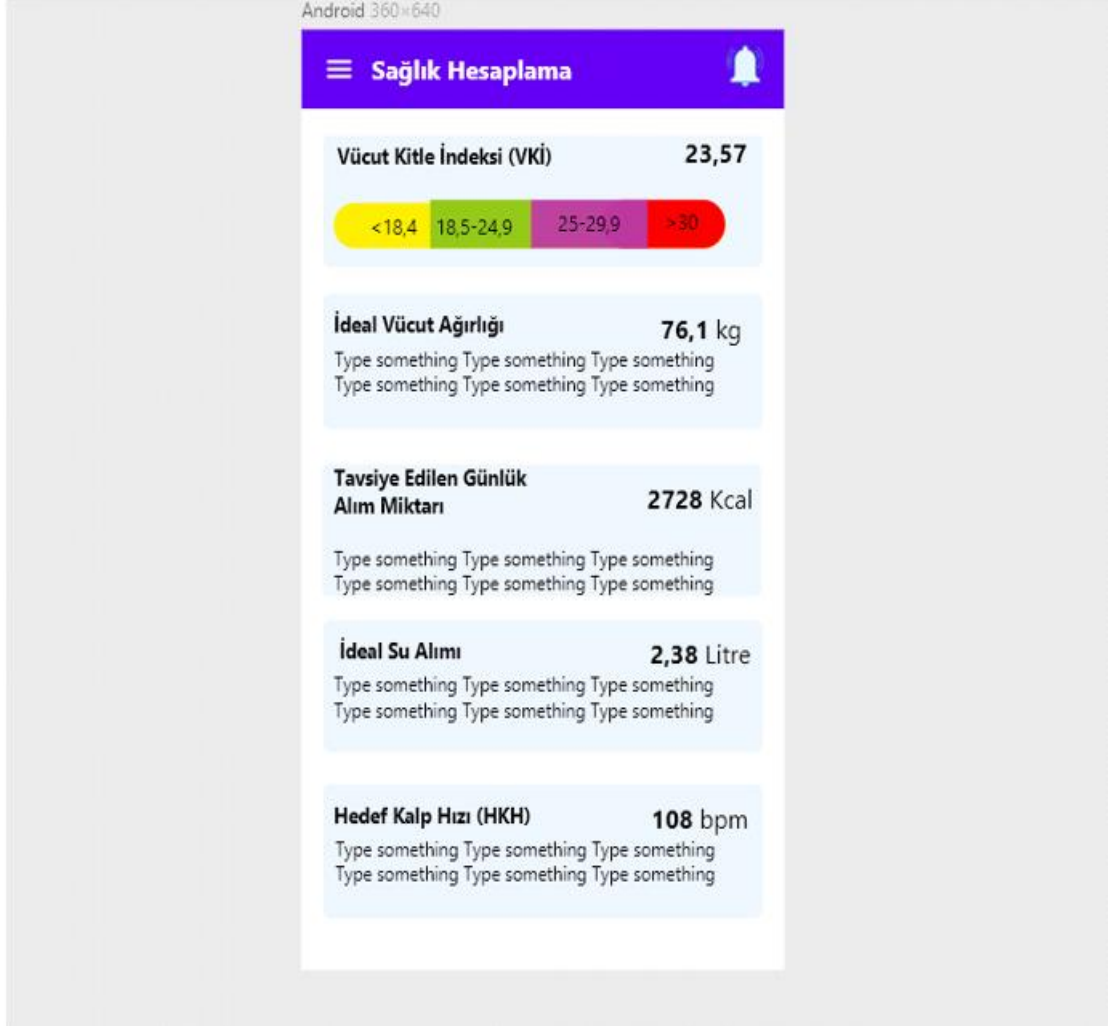
3. Ana Sayfa Kısmı



2.3. Ana Sayfa

Bu kısımda birden fazla buton ekranı görmekteyiz. Bu sayfada uygulamamızda yer alan ve bizim sağlığımız açısından uymamız gereken kurallar bu butonların içerik kısmında yer almaktadır. Uyku kısmında kullanıcının ne kadar uyuduğumuz ve daha ne kadar uyumamız gerektiği gibi veriler yer almaktadır.

4. Sağlık Hesaplama Kısmı



2.4. Sağlık Hesaplama

Sağlık hesaplama kısmında vücudumuza dair tüm hesaplamaların yapıldığı bölümdür. Bu hesaplamalara göre de en iyi idealleri ortaya koyan sayfamızdır. Vücudumuzda kontrol altında tutmak istediğimiz değerler de bu sayfanızda yer almaktadır.

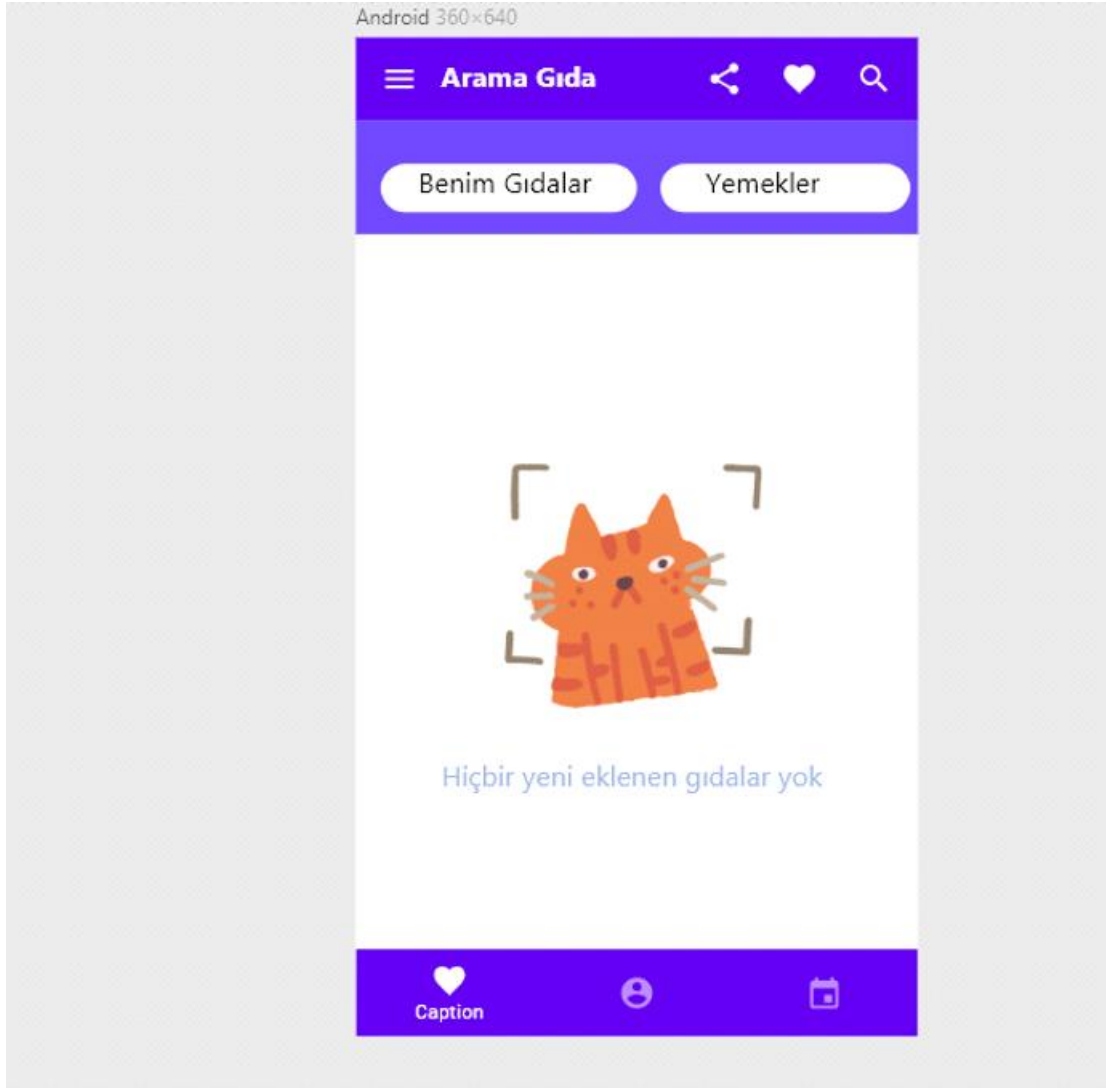
5. Günlük Sayfası



2.5. Günlük

Günlük sayfamızda o gün içerisinde vücudumuza almış olduğumuz gıdaların kalori hesabını ve içmiş olduğumuz su miktarını göstermektedir. Bu sayede vücudumuzun kalori ve su alımını dengede tutmaktayız.

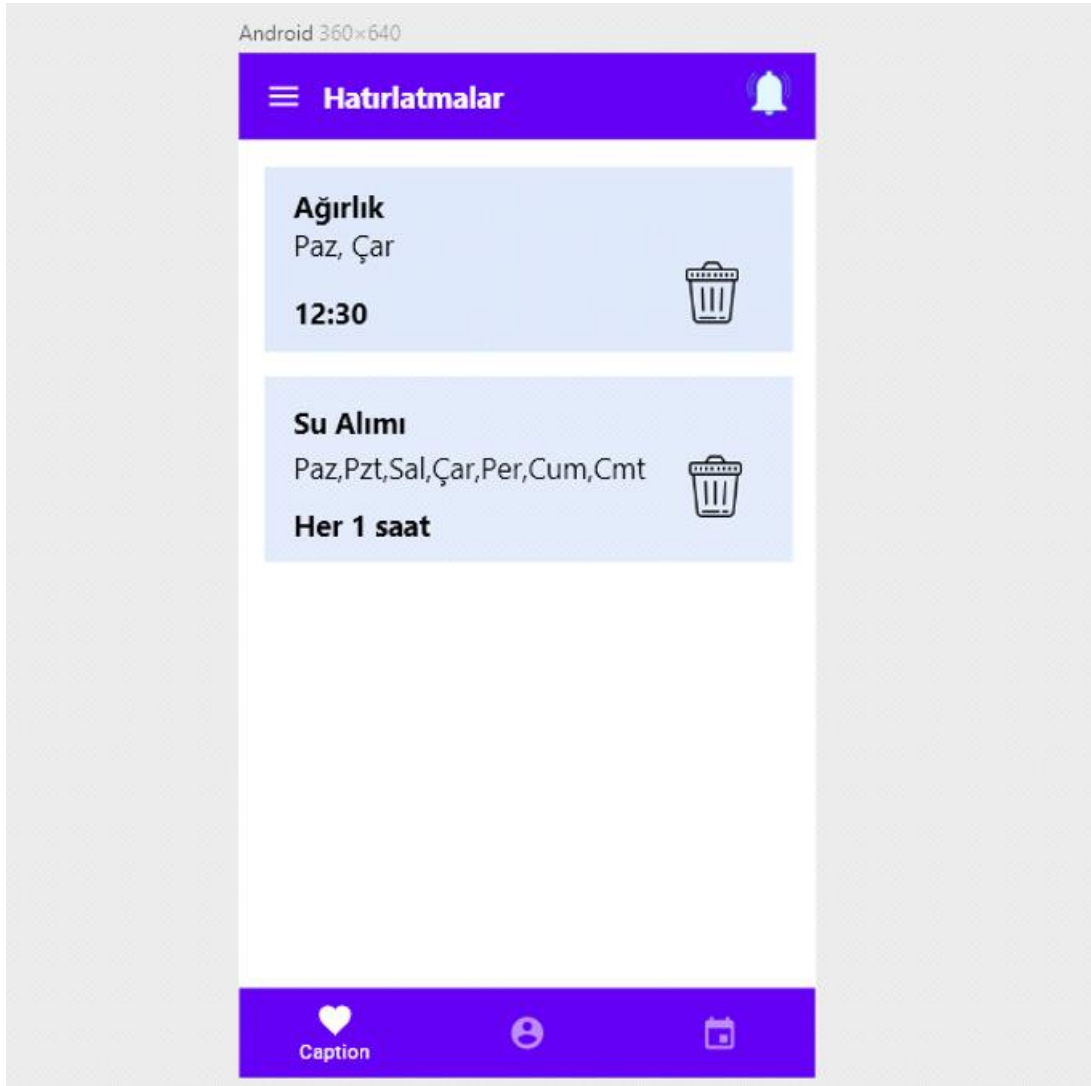
6. Gıda Arama



2.6. Gıda Arama

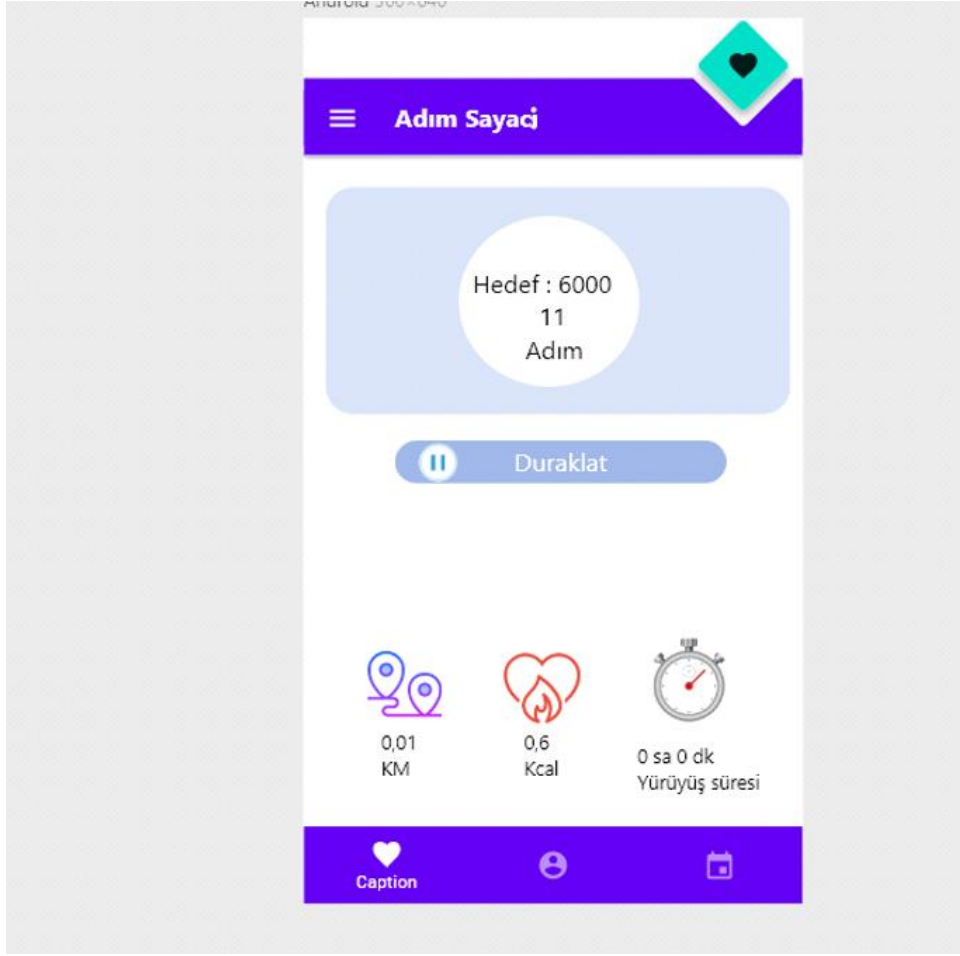
Gıda arama sayfamızda ise kaydedilen gıdaları tekrar arayabiliriz. Hatırlatmalar sayfamızda ise bir bireyin unutmuş olduğu sağlık aktivitelerini belli zamanlarda hatırlatarak vücudumuzun bir düzen içinde devam etmesini sağlamaktayız.

6.1. Hatırlatmalar



2.7. Hatırlatmalar

7. Adım Sayacı Ve Kalp Atım Hızı Ekranımız

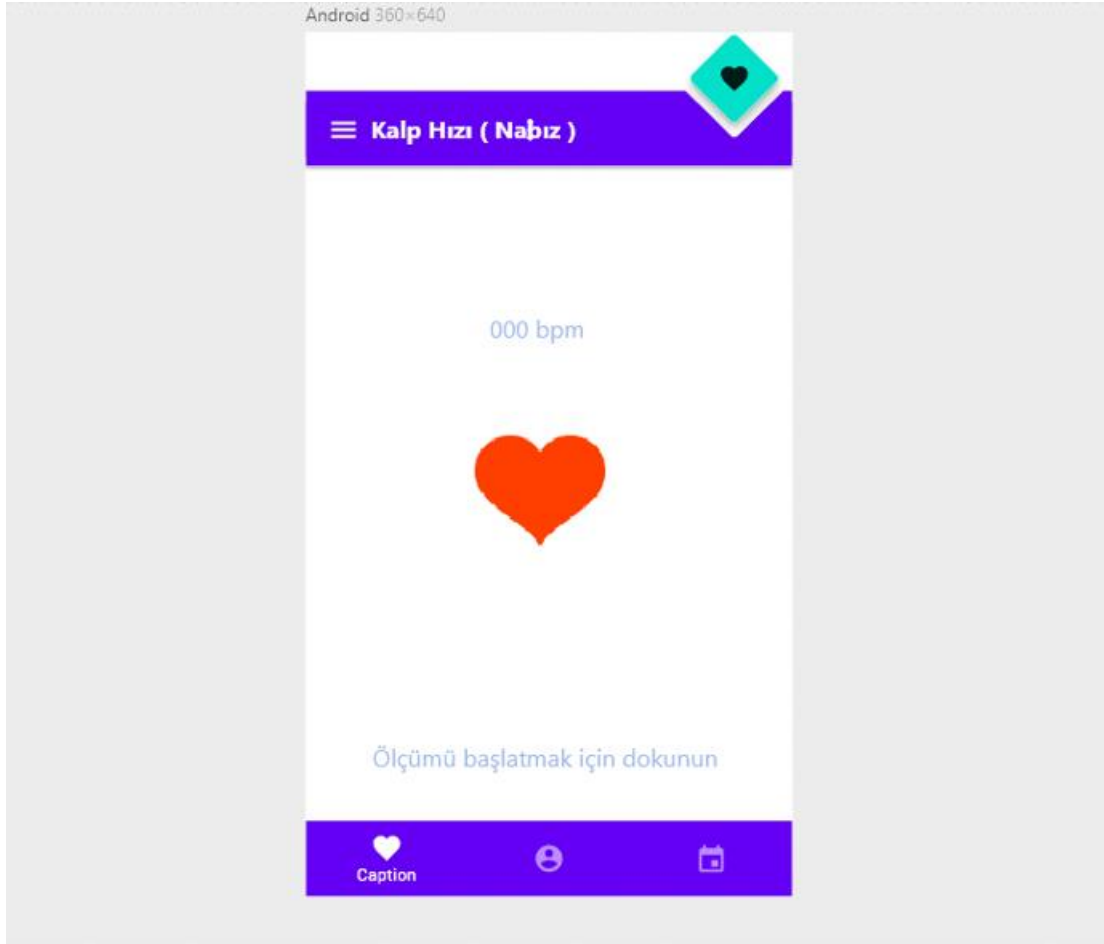


2.8. Adım Sayacı

Adım sayacımız 7/24 çalışmaktadır gün içerisinde atmış olduğumuz adımları, çıkmış olduğumuz merdivenleri, kat çıkarken attığımız adım sayılarımızı bize göstermektedir. Bunlarla beraber bu yürüyüşün bize ne kadar kalori yaktırdığı da kaydedilmektedir. Hedef adım sayısı sayesinde hedefimize yaklaşıp yaklaşmadığımızı görebilmekteyiz.

Kalp atım hızında ise bizim nabız sayımızı bize saatlik olarak ölçüp gösteren bir uygulama ekranımız. Bu uygulama bize herhangi bir anormal durumda uyarı verebilmektedir.

7.1 Kalp Atım Hızı (Nabız)



2.9. Kalp Atımı

6-)Testlerin Tasarımı

A. Sınıf Testleri

Amaç: Tüm ürünün temel işlevini test etmek için, aşağıdakileri içeren işlevleri test etmeyi planlıyoruz:

1. MySQL veri tabanının oluşturulması.
2. MySQL veritabanının uygulamadan çekilen verilerle bağlantısı ve verileri uygulamada görüntüleme
3. Uygulamanın temel görünümünü test edin.
4. Ana sayfa işlevini test edin

5. Sağlık Hesaplama Fonksiyonlarını test edin
6. Nabız İndeksi Ölçer fonksiyonunu test edin
7. Adım Sayacı Ölçer fonksiyonunu test edin
8. Uyku İzleme fonksiyonunu test edin
9. Gıda arama Fonksiyonunu test edin
10. Hatırlatma Analiz fonksiyonu test edin

Sonuçlar:

1. MySQL veritabanı bilgisayarımızda başarıyla oluşturulmuştur
2. Uygulamamızdan toplanan veriler MySQL veritabanında başarılı bir şekilde saklanabilir ve uygulamamızda başarıyla görüntülenebilir
3. Uygulamanın kullanıcı arayüzü ve menü kısmı gösterilebilir durumda.
4. Genel kilo, günlük su , edinilen kalori ve yakılan kalori ekranda gösterilebilir nabız ölçüm tasarlanma aşamasında.
5. Kullanıcının kilo ve boy verilerine göre vücut kitle analizi , ideal vücut ağırlığı , bazal metabolizma, ideal su alımı gösterilebilir .
6. Uygulamamızın nabız İndeksi Ölçer fonksiyonunu tasarım aşamasında.
7. Adım sayacı ölçer fonksiyonu hesaplanabilir ve ekranda gösterilebilir.
8. Uyku İzleme fonksiyonu hesaplanabilir ve ekranda gösterilebilir.
9. Gıda arama Fonksiyonunu hesaplanabilir ve ekranda gösterilebilir.
10. Kullanıcı belirlediği zaman ve saatte hedefini görüntüleyebilir ve hatırlatma uyarısı verilir.

B. Fonksiyonel Birim Testleri

Projemizin test senaryolarının tasarımları verilmiştir. Her test durumu şemada listelenmiştir.

Test1 Giriş

Giriş Gereksinimi	Beklenen Çıktı	Başarılı/ Başarısız	Yorumlar
Veri	Hesaplanmış Veri	Veriler eksiksiz ve doğru ise olumlu.	Bu test giriş içindir

7-)Proje Yönetimi ve Çalışma Planı

a. Ekip Üyelerinin Bireysel Katkıları

a.1 Rümeysa Taş

Algoritma ve veriler konusu üzerinde ilk başta bazı kaynaklardan araştırma yaptım . Genellikle bazı projelerde ortada dönen büyük ve tek algoritma olabiliyor lakin bizim projemizde kısa ve daha fazla algoritma var . Projemizin üzerinde tekrardan baştan sona derin bir inceleme yaptım . Projenin üstünde gerekli algoritmaları tek tek inceledim, hepsine matematiksel model oluşturdum . Bazılarına örnek kod ekledim , bazılarına akış diyagramı . Ekran fotoğraflarıyla hangi fonksiyonda hangi matematik model olduğunu belirttim. Test tasarımında ise oluşturmaya başladığımız projedeki gerekli test aşamaları ve sonuçlarını kayıt ettik. Çalışma planı vs. hepsi ortak olarak meet üzerinden gerçekleştirdiğimiz yayında planlayıp oluşturduk

a.2 Ayşe Çay

Sınıf diyagramları ve arayüz özellikleri ile sistem mimarisi ve sistem tasarımı başlıkları altında bulunan konuları Ayşe UYAR arkadaşım ile birlikte yazdım. Öncelikle çevrilerimizi yapıp araştırmalarımıza başladık. Kapsamlı bir araştırmanın ardından raporumuzu yazmaya başladık. Sınıf diyagramlarını yapmak için draw.io uygulamasını kullanıp ortaya güzel bir diyagram çıkardık. Daha sonrasında raporumuzun kemik kısmını tamamladık. Bu tamamlamanın ardından kapsamlı halde rapor yazımını yaptık. Yapıyor olduğumuz uygulamayı detaylandırıp arayüzünün özelliklerini tanımladık.

Sistem mimarisi ve tasarımı kısmımızda da tablolarımızı yine aynı uygulamadan yaptık. Raporumuzun bütün kısımlarının arkadaşım ile birlikte yaptım. Rapor kısmımızı da bu şekilde tamamladık.

a.3 Ayşe Uyar

Sınıf diyagramları ve arayüz özellikleri ile sistem mimarisi ve sistem tasarımı başlıkları altında bulunan konuları Ayşe ÇAY arkadaşım ile birlikte yazdım. Öncelikle çevrilerimizi yapıp araştırmalarımıza başladık. Kapsamlı bir araştırmanın ardından raporumuzu yazmaya başladık. Sınıf diyagramlarını yapmak için draw.io uygulamasını kullanıp ortaya güzel bir diyagram çıkardık. Daha sonrasında raporumuzun kemik kısmını tamamladık. Bu tamamlamanın ardından kapsamlı halde rapor yazımını yaptık. Yapıyor olduğumuz uygulamayı detaylandırıp arayüzünün özelliklerini tanımladık.

Sistem mimarisi ve tasarımı kısmımızda da tablolarımızı yine aynı uygulamadan yaptık . Raporumuzun bütün kısımlarının arkadaşım ile birlikte yaptım . rapor kısmımızı da bu şekilde tamamladık.

a.4 Recep Batuhan Günay

Etkileşim diyagramlarını yaparken her bir Use - Case için dizi diyagramlarını daha önceden çizmiş olduğum diyagramları kullanarak bunları raporda kullandım. Dizi diyagramların altına hangi Use-Case diyagramı olduğunu da ekledim. Açıklama kısmında yukarısında gösterilen dizi diyagramın ne işe yaradığını kısaca belirttim ve açıklamaların devamında başlatıcı aktörü, yardımcı aktörü, aktörün hedefini, önkoşulları, beklenen şartları ve başarılı bir senaryo için gidilecek adımları belirttim. Test tasarımında ise oluşturmaya başladığımız projedeki gerekli test aşamaları ve sonuçlarını kayıt ettik. Çalışma planı vs. hepsi ortak olarak meet üzerinden gerçekleştirdiğimiz yayında planlayıp oluşturduk.

a.5 Cankat Ateş

Görev dağılımının yapılmasında ve izlenecek adımların belirlenmesinde ekip lideri olarak görev aldım. Daha sonra 1. Bölüm olan etkileşim diyagramlarında düzeltmeleri ve yönlendirmeleri yaptım. Ekip arkadaşlarımın yapmış olduğu bölümlere çeşitli sebeplerden dolayı dahil olarak çalışmalarda bulundum. Ayrıca tüm dökümanların biraraya getirilmesi ve raporun oluşturulması görevimdi. İçindekiler, kaynakça, çalışma planını bizzat bireysel olarak yaptım. Arkadaşlarımın dökümanlarını toplayarak rapor 2 yi meet üzerinden sunum yaparak biraraya getirdim ve raporu oluşturdum.

b. Proje Koordinasyon ve İlerleme Raporu

Öncelikle geliştirmekte olduğumuz projenin belirlenen adımlarını, bireysel yeteneklerimiz doğrultusunda ekip üyelerimize eşit sorumluluk alacak biçimde görev dağılımını gerçekleştirdik. Sorumlu olduğumuz alanlarda ki ilerlememizi ve projeye dair bütün haberleşmelerimizi meet ve çeşitli sosyal platformlar üzerinden gerçekleştirdik. Mevcut ödev takvimimizin doğrultusunda zamanında raporumuzu tamamladık.

c. Çalışma Planı

Çalışma Planı Çizelgesi



d. Sorumlulukların Dağılımı

- Recep Batuhan Günay ve Rümeysa Taş testten sorumlu ekip olarak görev alacaktır.
Projenin paylaştırılan bölümlerine göre kodlama ve geliştirme de tüm ekip üyeleri görev alacaktır.
- Entegrasyonu ekip lideri olarak Cankat Ateş koordine edecek ve inceleyecektir.
- Projenin genel testini Rümeysa ve Batuhan yaparken, çalışmalara katkısı olması ve bireysel olarak kendisini geliştirmesi için herkes kendi bölümünün test aşamasında görev alacaktır.

8-) Kaynakça

- <https://app.diagrams.net/>
- 1- <https://bidb.itu.edu.tr/seyir-defteri/blog/2013/09/08/s%C4%B1n%C4%B1f-diyagramlar%C4%B1>
 - 2- https://en.wikipedia.org/wiki/Personal_health_application
 - 3- <http://web.firat.edu.tr/mbaykara/ytm5.pdf>
<https://app.diagrams.net/>
 - 4- www.pinterest.com
 - 5- www.appdesignserved.com
 - 6- tr.m.wikipedia.org
 - 7- www.anadolusaglik.org
 - 8- <https://translate.google.com/>
 - 9- Google Clasroom/UML Modelleme Notları