

# RAPOR 3

---

## Streaming on Demand CamViewer

Sürüm #2

Tüm telif hakları © 2017 Semih KIROĞLU'na aittir.  
Her hakkı saklıdır.

**Semih KIROĞLU**

14.05.2017

## İçindekiler

<b>1. Giriş</b>	<b>4</b>
1.1 Amaç	4
1.2 Kapsam	4
1.3 Tanımlar, Kısa adlar ve Kısaltmalar	4
1.3.1 Terim ya da Kısaltma	4
1.3.2 Tanım	4
1.4 Kaynaklar	4
1.5 Genel Bakış	5
<b>2. Genel Açıklama</b>	<b>5</b>
2.1 Ürün Perspektifi	5
2.2 Ürün Fonksiyonallıkları	5
2.3 Kullanıcı Karakteristikleri	5
2.4 Genel Kısıtlar	5
2.5 Varsayımlar ve Bağımlılıklar	5
2.6 Kullanılan Teknolojiler ve Tasarım Dilleri	6
<b>3. Özel Gereksinimler</b>	<b>7</b>
3.1 Dış Arayüz Gereksinimleri	7
3.1.1 Kullanıcı Arayüzleri	7
3.1.2 Donanım Arayüzleri	7
3.1.3 Yazılım Arayüzleri	8
3.1.4 Haberleşme Arayüzleri	8
3.2 Fonksiyonel Gereksinimler	9
3.2.1 Kamera Ekle	9
3.2.2 Kamera Görüntüle	9
3.2.3 Kamera Yönet	10
3.2.4 Mobil Uygulamadan MMS Üzerinde Oturum Açma	10
3.3 Kullanım Senaryoları	11
3.3.1 Kullanım Senaryosu #1	11
3.3.2 Kullanım Senaryosu #2	11
3.3.3 Kullanım Senaryosu #3	12
3.4 Sınıflar / Nesnelere	12
3.5 Fonksiyonel-Olmayan Gereksinimler	13
3.5.1 Performans (Performance)	13
3.5.2 Güvenilirlik (Reliability)	13
3.5.3 Kullanılabilirlik (Availability)	13

3.5.4 Güvenlik (Security).....	14
3.5.5 Bakımı yapılabilirlik (Maintainability).....	14
3.5.6 Taşınabilirlik (Portability).....	14
3.6 Ters Gereksinimler.....	15
3.7 Tasarım Kısıtları.....	15
3.8 Mantıksal Veritabanı Gereksinimleri.....	15
3.9 Diğer Gereksinimler.....	15
<b>4. Analiz Modelleri.....</b>	<b>16</b>
4.1 Ardışıl Diyagramlar (Sequence Diagrams).....	16
4.2 Veri Akış Diyagramları (Data Flow Diagrams) (DFD).....	16
4.3 Durum-Geçiş Diyagramları (State-Transition Diagrams) (STD).....	16
<b>5. Değişim Yönetim Süreci.....</b>	<b>17</b>
<b>6. Test Çalışmaları.....</b>	<b>18</b>
<b>7. Sonuç.....</b>	<b>20</b>
<b>A. Ekler.....</b>	<b>21</b>
A.1 Ardışıl Diyagramlar (Sequence Diagrams).....	21
A.2 Durum-Geçiş Diyagramları (State-Transition Diagrams) (STD).....	22
A.3 Ekran Görüntüleri.....	25

# 1. Giriş

## 1.1 Amaç

*Bu rapor, geliştirilme süreci tamamlanmak üzere olan söz konusu sistemin tam olarak kimlere, hangi amaca, nasıl ve ne şekilde hizmet edeceğini açıklamak ve sistemin tasarlanmış dinamiklerini, kullanılacağı altyapı ve teknolojileri açıkça beyan etmek amacıyla hazırlanmıştır.*

## 1.2 Kapsam

*Streaming on Demand (SoD) projesi; bir veya birden çok IP kamera, bir mobil uygulama (CamViewer Mobil) ve bir middleman sunucu ile üzerinde çalışan web tabanlı uygulamadan (CamViewer Web) oluşmaktadır. Proje, IP kameraların streaming özelliğini, kullanıcının özgürce kullanabilmesi amacıyla kısıtlar ve kameraların kullanıcıya On Demand mantığında akış sağlamasına imkân sağlar.*

## 1.3 Tanımlar, Kısa adlar ve Kısaltmalar

1.3.1 Terim ya da Kısaltma	1.3.2 Tanım
SoD	Streaming on Demand
IP	Internet Protocol
YGD	Yazılım Gereksinim Raporu
RTMP	Real Time Messaging Protocol
RTSP	Real Time Streaming Protocol
DVR	Digital Video Recorder
MMS	Middleman Server
RPi	Raspberry Pi
PoE S/I	Power over Ethernet Splitter/Injector
LAN/WAN	Local/Wide Area Network
HTTP/HTTPS	Secure/Hyper Text Transfer Protocol
F/P	Fiyat/Performans
URL	Uniform Resource Locator
NAT	Network Address Translation
PIN	Personal Identifier Number

## 1.4 Kaynaklar

*Bu doküman oluşturulurken “IEEE Std 830-1998: IEEE Recommended Practice for Software Requirements Specifications” şablonu referans alınmıştır.*

## **1.5 Genel Bakış**

*Bu, raporun projenin geldiği son noktaları da içerir ikinci düzenlenmiş halidir.*

## **2. Genel Açıklama**

### **2.1 Ürün Perspektifi**

*SoD projesi, halihazırda RTMP veya RTSP protokollerini kullanarak akış farklı marka/model olan ve RTMP/RTSP akışı sağlayan tüm IP kameraları tek bir noktadan görüntülemeyi ve yönetmeyi amaçlayan, DVR yapılarına kıyasla daha düşük maliyetlere çözüm sunan, modüler yapıda tasarlanan bir projedir.*

### **2.2 Ürün Fonksiyonallıkları**

*SoD projesinin, IP kameraların eş zamanlı izlenebilmesi, merkezi görüntüleme ve kontrol fonksiyonallıklarına sahip olacağı ön görülmektedir. Bu süreçlerde kameraların eş zamanlı ve merkezi görüntüleme özelliği web ve mobil uygulama üzerinden merkezi kontrol aracılığı ile MMS üzerinden kullanılabilir.*

### **2.3 Kullanıcı Karakteristikleri**

*SoD projesi, kurulum ve kullanım için temel düzeyde bilgisayar ve ağ yönetim bilgisi gerektirir. Bunun dışında ek olarak bir uzmanlık gerektirmez ancak hedef kitle olan son kullanıcının sıradan bilgisayar kullanıcısı olmaması öngörülmektedir.*

### **2.4 Genel Kısıtlar**

*Projenin planlanan hiçbir noktasında geçerli değildir.*

### **2.5 Varsayımlar ve Bağımlılıklar**

*SoD projesi, birbiriyle haberleşen (ve zorunlu olmayan ek ve faydalı) birçok modülden oluşan bir projedir. IP kamera, MMS ile web üzerinden kullanılabilir. Aynı zamanda mobil uygulama üzerinden sisteme bağlanıp taşınabilir halde kullanılabilir.*

## 2.6 Kullanılan Teknolojiler ve Tasarım Dilleri

*Frontend tarafta Őu teknolojilerden faydalanılmıŐtır:*

- *HTML*
- *CSS*
- *Ionic*
- *Angular*

*Backend tarafta Őu teknolojilerden faydalanılmıŐtır:*

- *PHP*
- *Ionic*
- *Angular*
- *VLC*
- *Bash*
- *Ubuntu Server (x86 veya x64 iin)*
- *(Veya ARM iin Raspbian)*

*Sistemin geneli OOP/MVC ile tasarlanmıŐtır. Bazı kısımlar ise function-based/OOP ikilisi koordineli biimde bir arada kullanılarak tasarlanmıŐtır.*

### 3. Özel Gereksinimler

#### 3.1 Dış Arayüz Gereksinimleri

##### 3.1.1 Kullanıcı Arayüzleri

*Kullanıcılar; projeye, iki farklı arayüzden etkileşime geçebilir. Planlanan ilk tasarımlara göre;*

*- (MMS üzerindeki) Web uygulaması tarafında, sisteme yeni kameraların eklenebileceği bir modal box ve ekranın geri kalanında kameraların thumb görüntüleri, kullanıcı tarafından belirlenmiş grid yapı üzerinde mevcut olacaktır. Kamera görüntülerine çift tıkladığında görüntü, yüksek çözünürlükte ve maksimum boyutta görüntülenecek, sağ tıkladığında veya ilgili kamera görüntüsü tek tıkladığında sağ üstte belirecek cogs ikonuna tıklayarak kameranın yönetimine dair işlemler ekranına ulaşılacaktır, aynı ekrandan kamera silme veya gücünü açıp/kapatma özelliklerine de erişilecektir.*

*- Mobil uygulama tarafında da yapı benzer olacaktır. Bu sayede aynı yapıya ait arayüzlerin tekrar tasarlanma maliyetinden kaçınılmış olunacak ve aynı zamanda kullanıcılar iki arayüzü farklı ürünlere aitmiş gibi düşünmeyecektir. Mobil uygulamada web uygulamasına ek olarak, middleman server'da dahil edilebilecek ve tüm kameralar tek bir merkezden yönetilebilir olacaktır.*

*Her iki arayüzün tasarımında da öncelikli olan husus kullanıcı deneyimi ve kullanıcının sistemden alacağı verim olacaktır. Örneğin; internet erişim hızı düşük olan veya kotalı internete sahip olan bir kullanıcının grid yapıda tüm kameralar görünürken harcayacağı trafiği minimize edecek alternatifler, kullanıcı arayüzü üzerinde, kullanıcı tarafından değiştirilebilir biçimde sunulacaktır.*

##### 3.1.2 Donanım Arayüzleri

*Sistem, IP kameraları kablolu ağ üzerinden, bir merkezde toplamayı amaçlamaktadır. Bu projede merkez olarak kullanılacak makine bir MMS olacaktır ve planlamalara göre öncelikli olarak Ubuntu Server ile çalışan x64 bir bilgisayar olacaktır. Küçük alanlarda az kamerayla ve düşük performansta çalışılacak durumlar için RPi türevi bir bilgisayar kullanılacaktır. Görüntüleme ve yönetim noktasında mobil cihazlar ve masaüstü/dizüstü bilgisayarlar kullanılacak olup, güç yönetimi noktasında ise (gerekirse) Arduino prototipleme kartları ve PoE S/I cihazlar kullanılacaktır.*

### **3.1.3 Yazılım Arayüzleri**

*Sistemin ilk geliştirilmesi; mobil uygulama için Android’de çalışacak şekilde, MMS için Debian/Ubuntu’da çalışacak şekilde, web uygulaması için ise tüm modern tarayıcılarda ve farklı çözünürlükteki cihazlarda çalışacak şekilde yapılacaktır.*

*Kameralardan gelen görüntülerin her platformda oynatılabilmesi için MMS üzerinde çalışması öngörülen kodek; video için VP80, ses için Opus olarak belirlenmiş olup, mux olarak WebM kullanılmıştır.*

*Web uygulaması için geliştirme, öncelikle HTML5, CSS3, JavaScript teknolojileri kullanılarak yapılacaktır. Mobil uygulama için geliştirme, öncelikle cross-platform çerçevesinde geliştirme maliyetleri düşünülerek hybrid (Ionic) ile yapılacaktır.*

*Veriler MMS üzerinde MySQL veritabanında saklanacaktır. Mobil uygulamanın çalışması ve MMS’e bağlanması için gereken veriler WebStorage üzerinde saklanacaktır.*

### **3.1.4 Haberleşme Arayüzleri**

*Sistem, LAN üzerinden çalışacaktır. Kullanıcılar ise öncelikle LAN üzerinden sistemle haberleşecek, ilerleyen süreçte başarılı olduğu takdirde WAN (GPRS, 2G/3G/4G, Wi-Fi, WiMax) üzerinden de haberleşebileceklerdir.*



## 3.2 Fonksiyonel Gereksinimler

### 3.2.1 Kamera Ekle

#### 3.2.1.1 Hatasız Süreç

- *Kullanıcı, mobil veya web uygulamasındaki kamera ekle menüsünü takip ederek ilgili ekrana ulaşır. İlgili ekranda öncelikli olarak kameraya ait RTMP/RTSP linklerinden biri ve kameraya verilecek isim istenir. Süreç sonlandığında ilgili kamerayla bağlantı sağlanır ve kamera bilgileri veritabanına kaydedilir.*

#### 3.2.1.2 Süreç Hataları

- *Kullanıcı, gerekli erişim protokollerine ve parametrelere sahip değildir. Sisteme erişemez.*
- *(Web için) Kullanıcının sisteme eriştiği tarayıcı JavaScript desteklemiyordur. Görüntüleme sorunu yaşayacağı için erişimi engellenecektir.*
- *Kullanıcının girdiği RTMP/RTSP linkleri geçersizdir ve kameraya ulaşamıyordur. Veritabanı kaydı iptal edilerek kullanıcıya bilgi verilir.*
- *Kullanıcı bir veya daha fazla zorunlu alanı boş bırakmıştır. Veritabanı kaydı iptal edilerek kullanıcıya bilgi verilir.*

### 3.2.2 Kamera Görüntüle

#### 3.2.2.1 Hatasız Süreç

- *Kullanıcı, mobil veya web uygulamasından görüntülemek istediği kamerayı seçer. Kameranın birincil akışına bağlanılır ve yeni görünüme geçilerek kamera görüntüleme başlar.*

#### 3.2.2.2 Süreç Hataları

- *Kullanıcı, gerekli erişim protokollerine ve parametrelere sahip değildir. Sisteme erişemez.*
- *(Web için) Kullanıcının sisteme eriştiği tarayıcı JavaScript desteklemiyordur. Görüntüleme sorunu yaşayacağı için erişimi engellenecektir.*
- *Kameranın RTMP/RTSP linklerine erişilemiyordur. Görüntüleme iptal edilerek kullanıcıya bilgi verilir.*

### **3.2.3 Kamera Yönet**

#### **3.2.3.1 Hatasız Süreç**

- *Kullanıcı, mobil veya web uygulamasından yönetmek istediği kamerayı seçer ve ilgili ekrana ulaşır. Bu ekran üzerinden kameranın adını düzenleyebilir, RTMP/RTSP linklerini değiştirebilir, (mümkünse) kameranın gücünü açıp/kapatılabilir ve kamerayı silebilir.*

#### **3.2.3.2 Süreç Hataları**

- *Kullanıcı, gerekli erişim protokollerine ve parametrelere sahip değildir. Sisteme erişemez.*
- *(Web için) Kullanıcının sisteme eriştiği tarayıcı JavaScript desteklemiyordur. Görüntüleme sorunu yaşayacağı için erişimi engellenecektir.*
- *Kameranin güncellenen RTMP/RTSP linklerine erişilemiyordur. Güncelleme iptal edilerek kullanıcıya bilgi verilir.*
- *Kullanıcı bir veya daha fazla zorunlu alanı boş bırakmıştır. Güncelleme iptal edilerek kullanıcıya bilgi verilir.*
- *Kamera, güç yönetimini destekleyen donanıma sahip değildir veya bu donanım entegre edilmemiştir. İşlem iptal edilerek kullanıcıya bilgi verilir.*

### **3.2.4 Mobil Uygulamadan MMS Üzerinde Oturum Açma**

#### **3.2.4.1 Hatasız Süreç**

- *Kullanıcı, mobil uygulamayı çalıştırdığı sırada MMS IP adresi, kullanıcı adı ve şifre sorulur. Süreç sonlandığında ilgili MMS'e bağlantı sağlanır ve kamera bilgileri MMS veritabanından okunur ve kullanılır.*

#### **3.2.4.2 Süreç Hataları**

- *Kullanıcı, gerekli erişim protokollerine ve parametrelere sahip değildir. Sisteme erişemez.*
- *Girilen MMS linkine erişilemiyordur. Kayıt iptal edilerek kullanıcıya bilgi verilir.*
- *Kullanıcı bir veya daha fazla zorunlu alanı boş bırakmıştır. Kayıt iptal edilerek kullanıcıya bilgi verilir.*

### 3.3 Kullanım Senaryoları

#### 3.3.1 Kullanım Senaryosu #1

*Bir ev kullanıcısının bu sistemi evinde, birden çok farklı marka/model kamera ile kullandığını varsayalım. Bu projeyi kullanmadığı taktirde en iyi ihtimalle bir DVR ile çoklu görüntüleme sağlayacak. Ancak universal yapıda çalışıp birden çok farklı marka/modeldeki IP kameraları bağlayacağı DVR çeşitleri, f/p dengesi gözetilirse en fazla 8 kamera desteklemektedir. En kötü ihtimalle ise her üreticinin kendi mobil uygulamasını veya web arayüzünü kullanarak, kameraların URL'lerini saklayarak ayrı ayrı görüntüleme yapabilecek, bir noktadan kolayca izleme yapamayacaktır.*

*Sahada birden çok farklı marka/model IP kameranın kullanıldığı bir senaryoda SoD projesini kullanan kullanıcılar öncelikle mahremiyet gizliliğini bir nebze daha fazla sağlayabiliyor olacak (güç yönetimi, projede sağlanmış olduğu taktirde) ve sistemi merkezi olarak, marka/model gözetmeksizin görüntüleyip yönetebilecektir. Bunun yanı sıra bağlanılacak IP kamera sayısını teorik olarak yalnızca bant genişliği kısıtlayabilecektir. Ancak ev kullanımı için projenin tasarımında planlanan 100Mbps bant genişliğinin yeterli olacağı öngörülmektedir.*

#### 3.3.2 Kullanım Senaryosu #2

*Evinde Alzheimer hastalığına sahip bir yakını yaşayan ev kullanıcısının bu sistemi evinde, birden çok farklı marka/model kamera ile kullandığını varsayalım. Bu projeyi kullanmadığı taktirde kameralar NAT arkasında kalacak ve dışarıdan izlenme olasılığı olmayacaktır. NAT önüne alınan kameralar da sürekli akış sağlayacak ve (kameraların bazı modellerindeki kullanıcı/parola istemi yok sayılırsa) IP adresi belirlendiği taktirde herhangi bir kimse tarafından izlenebilir olacaktır.*

*Projenin kullanılması halinde evdeki kameraların her biri için NAT önüne alınması şartı ortadan kalkacak (bunun için ya MMS başka bir merkeze tünellenecek veya sadece lokal MMS NAT önüne alınacak) ve merkezi yollardan izlenebilecektir. Bunun yanı sıra (güç yönetimi için gerekli donanım kurulduğu taktirde) doğrudan izlemeye kapalı olacak, yalnızca ev sahipleri gerektiğinde kameraya güç sağlayarak kamerayı devreye alacaklardır.*

### **3.3.3 Kullanım Senaryosu #3**

*Bir veya birkaç tane dükkâna sahip bir esnaf/zincir mağaza sorumlusu olduğunu varsayalım. Sadece bir noktadan yayını transcode edecek şekilde diğer mağazalardaki farklı marka/model IP kameraları NAT önüne çekerek transcode yapılan merkeze web/mobil uygulama ile erişerek sistemi zahmetsizce kullanarak kameraları görüntüleyebilir. Sistem, bu durumda, kullanıcının her ofisi için ayrı ayrı DVR (vb. ürünü) satın alıp karma platformlardan izlemesinin önüne geçerek işleri tek bir merkeze toplamak suretiyle kolaylaştırır.*

*Akış güvenliği ve NAT durumu gözetilerek kameralar tüm noktalardan basit yollarla tünellenerek merkeze akış gönderebilir. Bu durumda hem NAT arkasına alınma durumu kolaylaşır, hem de şifrelenmiş özel bir tünel üzerinden giden akışlar, merkez sunucu dışında herhangi bir kimse tarafından çözülemez/dinlenemez.*

### **3.4 Sınıflar / Nesnelere**

*Projede bu alanla ilgili herhangi bir veri mevcut değildir.*

## 3.5 Fonksiyonel-Olmayan Gereksinimler

### 3.5.1 Performans (Performance)

*SoD projesinin tahmini performans fikirleri şöyledir:*

- *Mobil uygulama için minimum performans gereksinimi Android 4 çalıştıran mobil cihazlar olacaktır.*
- *Web uygulama için minimum performans gereksinimi modern ve güncel tarayıcıları sorunsuz çalıştıran tüm bilgisayarlar olacaktır.*
- *MMS için minimum performans gereksinimi tüm kodikleri çalıştırabilen, minimum 100Mbps ile lokal ağa bağlanabilecek, x86 ve x64 işlemciler söz konusu olduğunda Ubuntu, ARM işlemciler söz konusu olduğunda Raspbian işletim sistemine sahip herhangi bir bilgisayar olacaktır.*
- *Kameraların eş zamanlı olarak MMS'e bağlanacağı sayı ortalama 16 olarak hesaplanmıştır.*
- *Kameraların doğrudan mobil cihazlara yayın göndermesi durumunda cihazın donanım özellikleri doğrudan etmen olacağı için, optimum bir tespit yapılamamıştır.*

### 3.5.2 Güvenilirlik (Reliability)

*Sistem en az, tek başına çalışan IP kameralar kadar güvenli olacaktır. Daha güvensiz bir sistem mümkün değildir.*

### 3.5.3 Kullanılabilirlik (Availability)

*Sistem temel düzeyde bilgisayar ve network bilgisine sahip her insanın kolayca kullanabileceği dinamikler üzerine tasarlanmıştır. Bir IP kamerayı kurup gelişmiş biçimde her metodu ile kullanabilen herkes kolayca sistemi kullanabilecektir.*

### 3.5.4 Güvenlik (Security)

*Sistemin tüm verileri base64 ve plain-text biçiminde saklanacaktır. Kontrolden geçememiş kimseler bu verilere erişim sağlayamayacak olup, yalnızca kimliğini sistem düzeyinde doğrulamış kullanıcılar erişebileceklerdir. Kullanıcı kimlik bilgileri ise sha1 hash'leme yöntemi kullanılarak saklanacaktır.*

*Kameralara; kimliğini sistem düzeyinde doğrulayamamış kimseler, kesinlikle erişemeyeceklerdir. Mobil uygulama üzerinden sisteme erişecek kullanıcılar yalnızca çıkış yaptığı veya bir şekilde uygulama verileri sıfırlandığında sisteme yeniden şifre ile erişeceklerdir. Diğer durumlarda sunucu ile mobil cihaz arasında token anlaşması sağlanacak, eğer token ve cihaz kimliği eşleşmezse kullanıcıdan tekrar şifre istenecektir. MMS sunucu ise kamera erişimleri dışında yalnızca 80 ve 443 portlarından gelen isteklere yanıt verecek, diğer portlardan gelen erişim isteklerini reddedeceklerdir. Kamera erişimleri ise kameraların sunucuda kaydedildiği ID'lere göre (en fazla 999 olacak şekilde) 8001-8999 portları üzerinden akış verecektir. İzleme isteği yapılmadığı takdirde akış sunucu tarafından kapatılacaktır. Yani doğrudan bu portlar, sürekli akış izlemeye müsaade etmeyecektir.*

### 3.5.5 Bakımı yapılabilirlik (Maintainability)

*Sistemin tamamı, mümkün olduğunca, ortak kullanılan fonksiyonların ayrı ayrı barındırılmadan kullanılmasına olanak sağlayacak biçimde tasarlanmıştır ve MVC mimarisini mümkün olduğunca kullanacaktır. Bu nedenle herhangi bir düzenleme yapıldığında, zaman maliyeti düşük olacak biçimde tüm projede değişecektir.*

*Güncellemelerin dağıtımları ise tüm platformlarda otomatik gerçekleşecektir ve kullanıcıların bu hususta çaba sarf etmemesi öncelikli olacaktır.*

### 3.5.6 Taşınabilirlik (Portability)

*Sistem öncelikle kur ve çalıştır mantığını benimsediği için taşınabilirlik hususu pek olası görülmemektedir. Ancak sistemin başka bir sahaya taşınması durumunda kurulum süreci zor olmayacaktır. Özetle; önerilmemektedir ancak kolayca gerçekleşmesi mümkündür.*

*Şayet sistemlerin yazılımlarının taşınabilirliğinden söz etmemiz gerekirse; mümkün mertebe cross-platform yapıyı destekleyeceğinden herhangi bilgisayar/mobil cihaz üzerinde kolayca kullanılabilir tasarlanmıştır.*

### **3.6 Ters Gereksinimler**

*Projede geçerli değildir.*

### **3.7 Tasarım Kısıtları**

*Sistem esnek donanımlarda kolayca çalışabilmeyi hedeflediği için şimdilik bu husus geçerli değildir. Ancak öncelik; mobil için Android, MMS için Debian/Ubuntu, web için JS olacaktır.*

### **3.8 Mantıksal Veritabanı Gereksinimleri**

*Sistemin tüm noktalarında veritabanı kullanılacak olup, kullanıcı doğrulama evresinden sistemin kullanımına kadar her anda projeye entegre olacaktır. Projenin tasarımına göre veritabanı plain-text düzeyinde veri saklayacak, bu nedenle depolama kapasitesi projenin tamamındaki gereksinimlere göre en küçük olan kısım olacaktır.*

### **3.9 Diğer Gereksinimler**

*Yoktur.*

## **4. Analiz Modelleri**

### **4.1 Ardışıl Diyagramlar (Sequence Diagrams)**

*Ek olarak A.1'de yatay düzende gösterilmiştir.*

### **4.2 Veri Akış Diyagramları (Data Flow Diagrams) (DFD)**

*Proje için geçerli değildir veya yeterli veri yoktur.*

### **4.3 Durum-Geçiş Diyagramları (State-Transition Diagrams) (STD)**

*Ek olarak A.2'de yatay düzende gösterilmiştir.*



## 5. Deęişim Yönetim Süreci

*Proje üzerinde yapılacak güncellemeler için geliştirici projenin gidişatına ve durumuna göre yeni özellikleri planlar ve aşama aşama uygular. Gerekli görülürse güncellemenin gereklilięi kamuoyuna sorarak planlamaya yön verilebilir. Geliştirici, alınan kararlar neticesinde yapılacak güncellemeyi YGD üzerine planlı ve sistematik bir çalışma yürütebilmek için ekler ve güncelleme sonucunda meydana gelebilecek hata ihtimallerini düşürmeye çalışır. Güncelleme yapılacak konumda görsel çalışma gerekliyse önce tasarım çizimleri tamamlanır ve kullanıcı arayüz tasarımı kodlanır.*

*Geliştirici çalışmasını tamamladıktan sonra (varsa proje test uzmanları, yoksa geliştirici) projeyi farklı ihtimalleri düşünerek test eder ve hataları bulup süreci tekrar döndürür. Bu kısa döngü sonucunda tespit edilmiş ve öngörölmüş tüm hatalar kapatılmışsa güncelleme sisteme eklenir ve güncellenen kısma (BETA) ibaresi ile geliştiriciye hataları rapor edecekleri iletişim kanalına yönlendiren bir ekran eklenir. Bu iletişim kanalına iletilen bildirimler değerlendirilir ve döngü tekrar başlar.*

*BETA sürecinin güncelleme büyüklüğüne ve kapsamına göre minimum 2 hafta olacak şekilde sürmesi öngörölür ve bu süreçte sistem sürekli test ve analiz edilerek hataların tamamının kapatılması üzerine çalışılır. Bildirilen ve tespit edilen hataların büyük çoğunluğunun kapatıldığı kararı alınırsa ilgili güncellemeyle eklenen BETA ibaresi ve iletişim kanallarına yönlendiren link kaldırılır. Güncellemede hala hata bulunursa kritik kapsamda hızlı güncellemeler yapılır ve yayınlanır. Döngü sonlanır.*

## 6. Test Çalışmaları

*Sistemin tamamı aşama aşama test edilmiş olup, her aşamada hatalar kontrol edilmiş ve çalışırılığı doğrulanmıştır. Sistemin büyüklüğü ve vaktin darlığı nedeniyle hiçbir yapıda unit test vb. uygulanmamıştır.*

*Sistemin web dışında kalan tüm modüller aynı anda geliştirilmiş ve test edilmiştir. Sonradan eklenen veya geliştirmenin devamı olarak eklenen her modül, önce kendi çalışırılığı test edilerek projeye dahil edilmiş, daha sonra proje üzerinde diğer modüllerle bir arada test edilmiştir.*

*Test aşamaları projenin şekillenmesinde ciddi rol oynamış olup, alacağı son hali yine bu testlere göre belirlenmektedir. Nedeni ise açık bir şekilde teorik bilginin her zaman pratikle uyuşmaması durumu ve hızla gelişen teknolojiler kullanılması ile alakalı, yenilenmeyen dokümanlar kullanılarak geliştirmenin ve pratik simülasyonunun yapılmasıdır.*

*Bu nedenle teorik bilgiyle ortalama bir yol alıp pratikte çalışmaları test ederek proje şekillendirilmiştir. Aşağıda yapılan bazı test çalışmaları ve projeye katkıları özetlenmiştir:*

- *Transkod performansı RPi üzerinde yeterli olmadığı gerekçesiyle değiştirildi.*
  - *Teorik hesaplamalar sonucunda RPi'in birden çok kamerayı performans düşmeden ve milisaniyelik gecikmelerle transkod edeceği hesaplanmıştı.*
  - *Pratikte yapılan testler sonucunda Ubuntu Server yüklü VM ile çalışmaya devam edilmesi kararına varıldı.*
- *Mobil VideoView kodek desteği sorunu nedeniyle ve geliştirme sürecinin hızlanması amacıyla hybrid geliştirme süreci başladı.*
  - *Android Native uygulamalarda bulunan VideoView yapısı RTMP akışları desteklemiyordu.*
  - *Android üzerinden çalışması için iki ihtimal vardı:*
    - *Android uygulamaya kütüphane dahil etmek. (Web uygulamasında çözümü yok)*
    - *Akışı transkod etmek. (Web uygulamasında da son teknolojilerle ilerlemek için gerekli)*
  - *Bu nedenle transkod şartı getirip tüm uygulamaları bir ana sunucuya bağlama zorunluluğu geldi.*

- *Mobil WebView/PC Video elementin kodek uyumsuzluğu nedeniyle kodek deęiştirildi.*
  - *Başta belirlenen mp4 kodek seti yavaş çalışan ancak etkili ve yüksek çözünürlükte video/ses akışı sunan bir kodekti.*
  - *Ancak Mobil WebView video elementiyle birlikte doğrudan mp4 akışı desteklemiyordu. Dokümanlar ise bu konuda karmaşık ve hatalıydı.*
  - *Test ederek en doğru kodek bütünüünün mux' u WebM olacak şekilde video için VP8, audio için Opus olduğu belirlendi.*
  - *İlgili yapı diğerlerine göre çok daha yavaş çalışıyordu ancak tüm platformlara uyumlu olduğu için kullanılmasına karar verildi.*

## 7. Sonu

*Proje, yeterli sre bulunamadığı iin maalesef ilk planlandığı Őekilde tamamlanmamıştır. Ancak uygulamanın misyonlarından biri olan “evrensel kablosuz kamera izlencesi” misyonu başarıyla tamamlanmıştır.*

*İlerleyen srete projeye yeni misyonlar eklenerek daha fonksiyonel, daha performanslı ve daha kullanıcı dostu bir sistem kurgulanabilir. Ancak pazarlama noktasındaki başarısı, senaryoların daha müşteri dostu ve maliyeti dŐk olması halinde olabilir.*

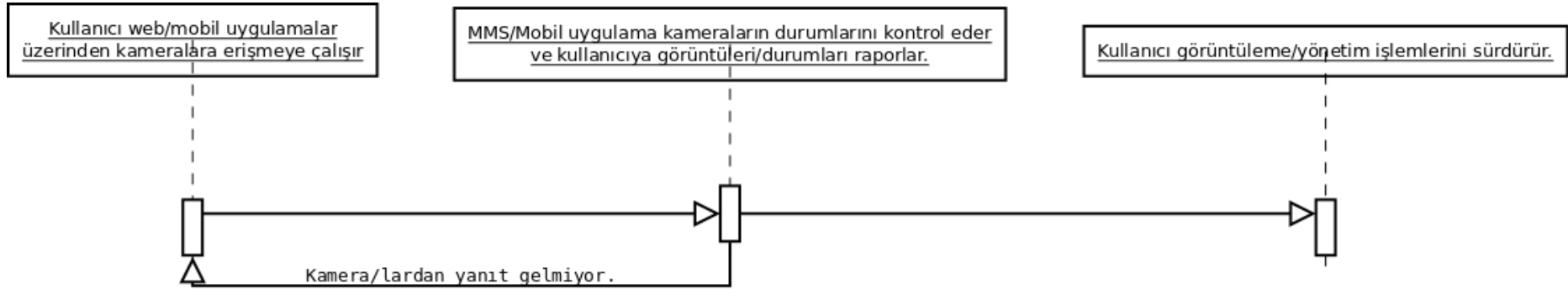
*Kişisel sonuçlarından bahsedecek olursak; bu proje şahsıma zaman yönetimi, algoritma tasarımı, farklı platformların entegre çalışması, yeni kodlama dilleri, ađ yönetimi, video/ses akışları gibi birçok konuda bilgi ve tecrbe katmıştır.*

*Bu srecin başından sonuna kadar destek oldukları iin başta ailem ve yoldaşım olmak zere, proje danışmanım Sn. Yrd.Do.Dr. Abdullah SÖNMEZ’e teŐekkrlerimi bor bilirim.*

*Ayrıca eđitim hayatım boyunca bize zerre katkıda bulunmak iin abalayan tm hocalarıma, eđitim/đretim hayatım boyunca bana destek olan ve gvenini hi esirgemeyen aileme Őkranlarımı bir bor bilirim.*

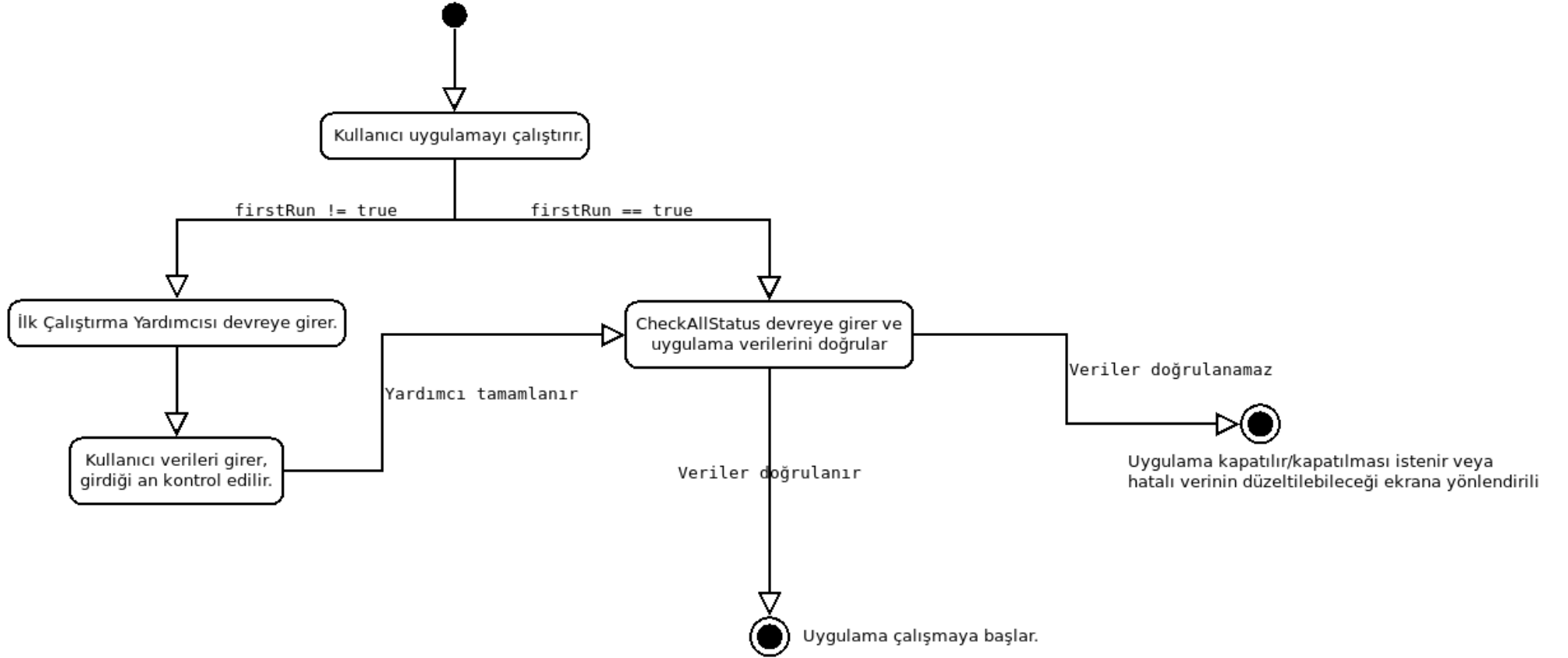
## A. Ekler

### A.1 Ardışıl Diyagramlar (Sequence Diagrams)

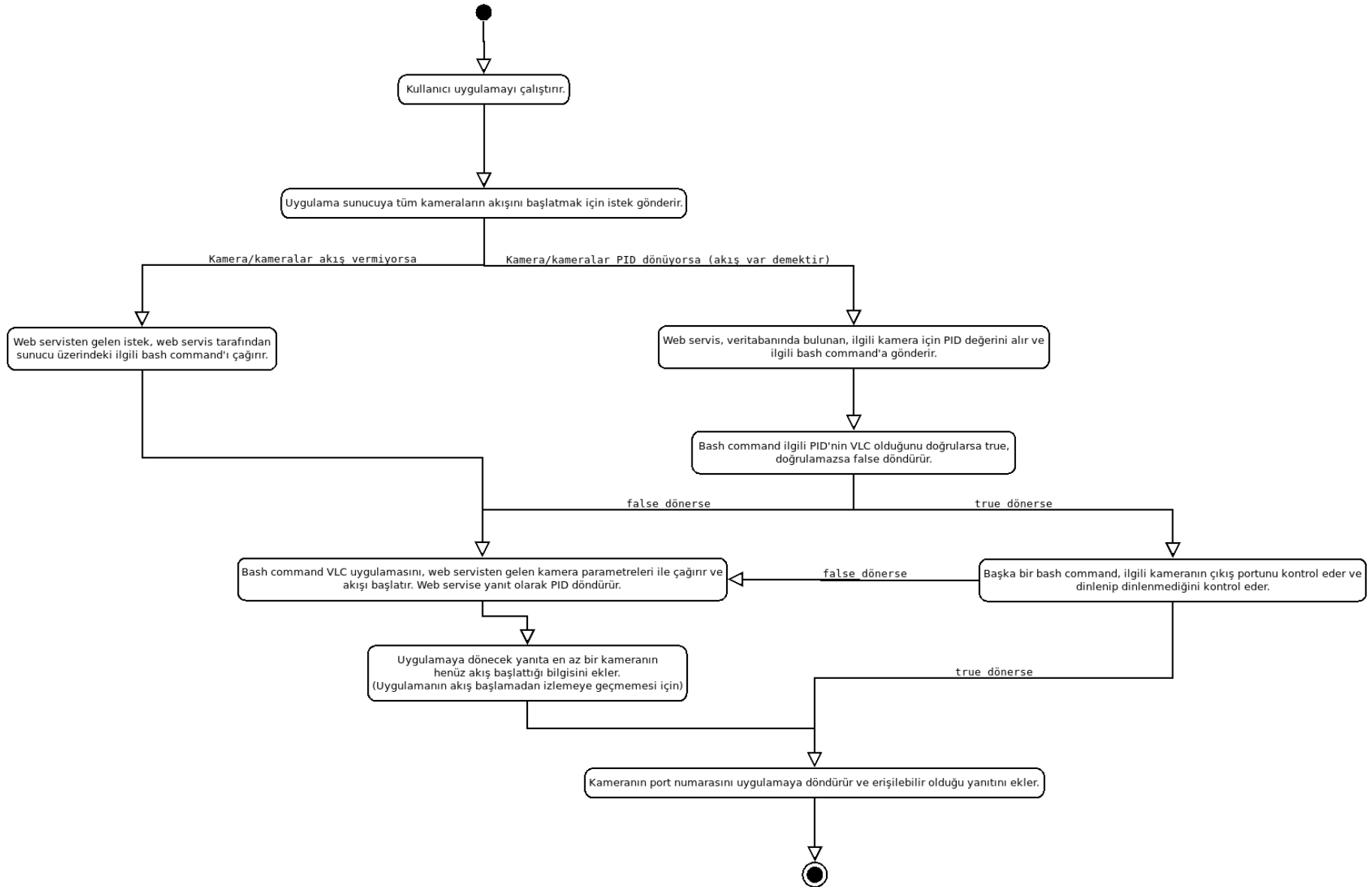


## A.2 Durum-Geçiş Diyagramları (State-Transition Diagrams) (STD)

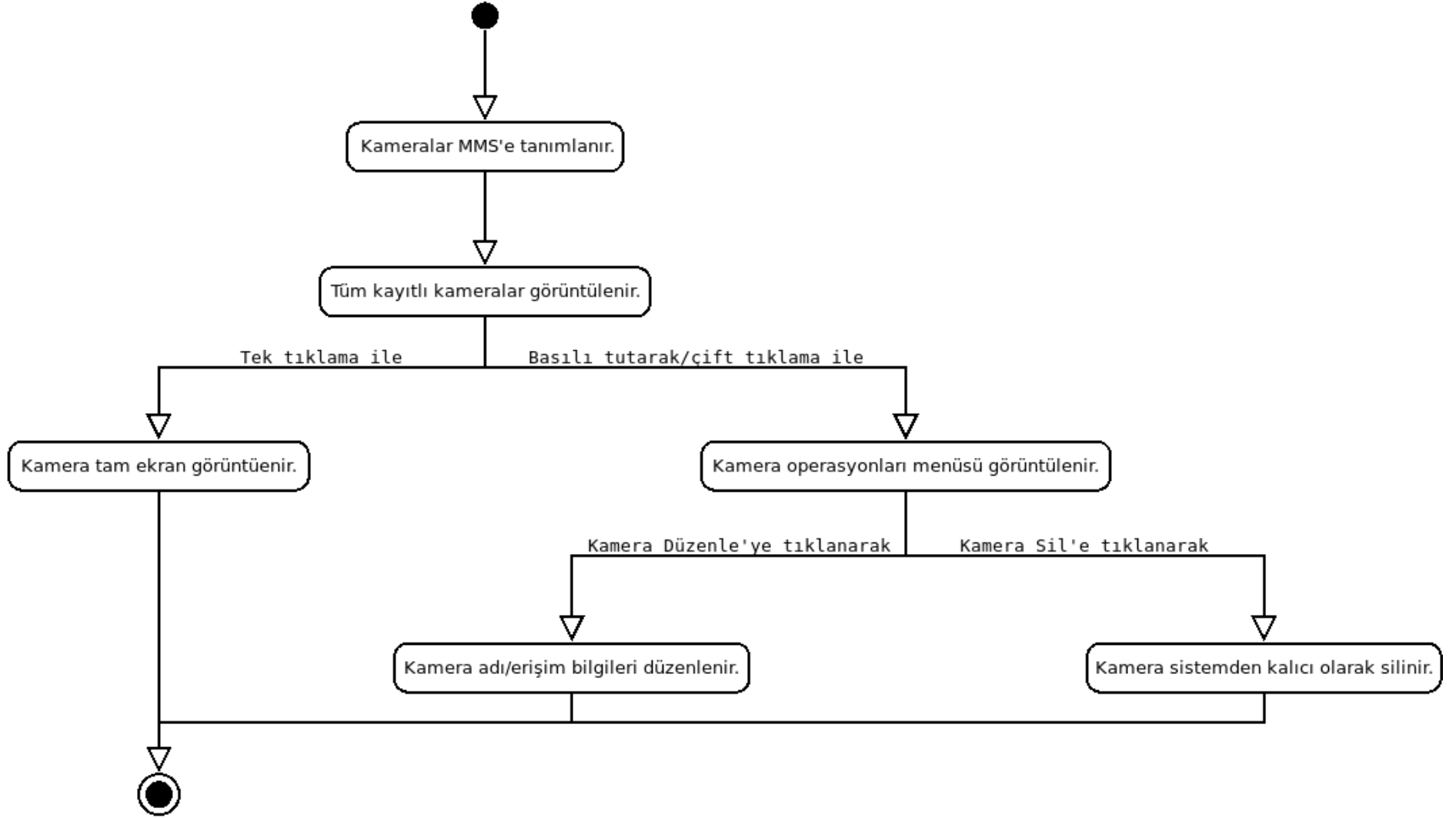
Uygulama Çalıştırma Diagram - Web/Mobil



Kamera İstekleri Diagram - Web/Mobil

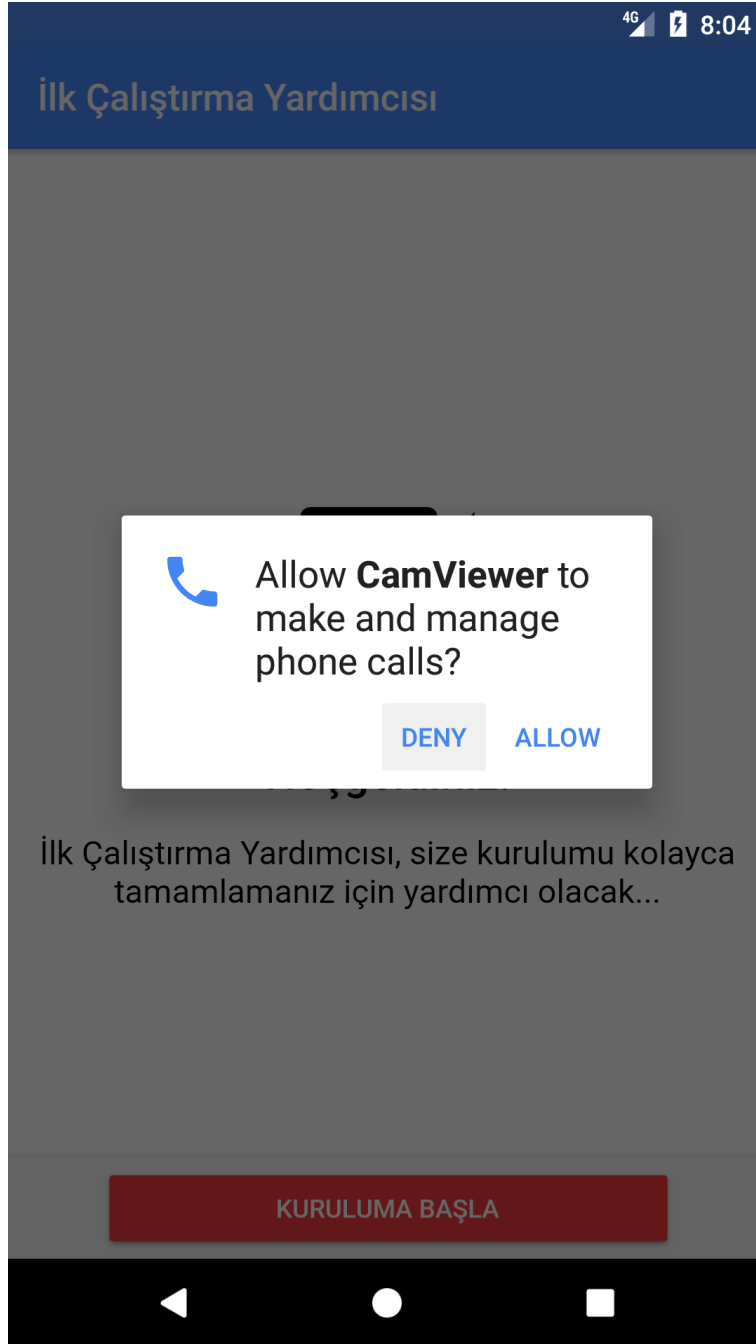


Ana Kamera Operasyonları - Web/Mobil





### A.3 Ekran Görüntüleri





## CamViewer Uygulamasına Hoşgeldiniz!

İlk Çalıştırma Yardımcısı, size kurulumu kolayca  
tamamlamanız için yardımcı olacak...

KURULUMA BAŞLA



## İlk Çalıştırma Yardımcısı



### CamViewer Sunucu Bilgilerini Girin

IP Adresi veya Makine Adı

192.168.1.39



SUNUCUYA BAĞLAN



## İlk Çalıştırma Yardımcısı



### CamViewer Kullanıcı Bilgilerinizi Girin

Kullanıcı Adı

semih

Şifre

•

OTURUM AÇ

## İlk Çalıştırma Yardımcısı



Hoşgeldin, Semih Kiroğlu

### CamViewer Pencere Yerleşimini Seçin

Izgara



Liste



AYARLA



### CamViewer Pencere Yerleşimini Seçin

Izgara

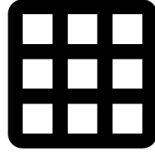


Liste



AYARLA





## CamViewer Izgara Yerleşimini Ayarlayın

BU NE DEMEK?

Izgara Sayısı

1 ▾



AYARLA



Izgara Sayısı

- 1
- 2
- 3
- 4
- 6

VAZGEÇ TAMAM



AYARLA



## İlk Çalıştırma Yardımcısı



### CamViewer Kurulumu Tamamlandı!

Tebrikler! CamViewer İlk Çalıştırma Yardımcısı'nı başarıyla tamamladınız. Artık sunucunuza kolayca kamera ekleyebilir, eklenmiş kameraları görüntüleyebilir ve yönetebilirsiniz.

BAŞLA

... Kontrol ediliyor...





••• Akış için 5 saniye bekleniyor...

Hoşgeldiniz, Semih Kirođlu





