

# APP INVENTOR İLE MOBİL UYGULAMALAR

---

AVCI UFUK YILMAZ | GÖKAY YILMAZ

# İÇİNDEKİLER

<b>BÖLÜM 1: APP INVENTOR</b>	<b>1</b>
Kavramlar	2
Algoritma	2
App Inventor Nedir?	5
GMail Hesabı Nasıl Açılır?	6
Nasıl Giriş Yapılır?	8
App Inventor Uygulamaları Nasıl Çalıştırılır?	10
Emülatör Kurulumu Nasıl Yapılır?	11
Mobil Aygıtlara Kurulum Nasıl Yapılır?	15
Neler Öğrendik?	17
<b>BÖLÜM 2: MENÜLER VE TEST</b>	<b>19</b>
Menüler	20
Projects Menüsü	20
Connect Menüsü	21
Build Menüsü	22
Help Menüsü	22
User Interface	24
Sağ Üst Menü	24
Layout	26
Media	26
Drawing And Animation	27
Maps	28
Sensors	28
Social	29
Storage	29

Lego Mindstorms	30
Connectivity	30
Sağ-Tık Menüleri	31
Yakınlaştırma ve Uzaklaştırma	33
Karar-Şart Yapısı	33
Döngüler	36
For Döngüsü	36
While Döngüsü	37
Do While Döngüsü	38
Foreach Döngüsü	38
Matematiksel İfadeler	39
Aritmetik Operatörler	39
Karşılaştırma Operatörleri	39
Mantıksal Operatörler	40
Neler Öğrendik?	41
<b>BÖLÜM 3: UYGULAMALAR</b>	<b>43</b>
Midillimi Seviyorum	44
Hayvan Seslerini Öğreniyorum	52
Piyanom	61
Konuşanları Yazıyorum	70
Havuçları Topluyorum	82
Bilgi Yarışması	93
Şu anda Ders Çalışıyorum	104
Çizim Programı Yapıyorum	111
Pedometre (Adımsayar) ile Kalori Hesaplama	118

Geri Dönüşüm Oyunu	123
WebViewler	128
Neler Öğrendik?	130
<b>BÖLÜM 4: UYGULAMAMI YAYINLIYORUM</b>	<b>131</b>
Google Play Console Nedir?	132
Google Play Console Üyelik İşlemleri	132
Keystore İşlemleri	134
Uygulamamı Yayınlıyorum	136
Son Söz	148

# 1

## APP INVENTOR

### BU BÖLÜMDE

Kavramlar	2
Algoritma	2
App Inventor Nedir?	5
GMail Hesabı Nasıl Açılır?	6
Nasıl Giriş Yapılır?	8
App Inventor Uygulamaları Nasıl Çalıştırılır?	10
Neler Öğrendik?	17

Bu bölümde ilk olarak bilmemiz gereken kavramlara değineceğiz. Yazılımın temeli olan algoritma mantığını öğreneceğiz. Ana konumuz olan App Inventor hakkında detaylı bilgi edinip üyelik işlemlerimizi tamamlayacağız.

App Inventor'un oldukça eğlenceli bir platform olduğunu ve bu yüzden uygulama geliştirmek için heyecanla beklediğinizi biliyoruz.

Yaptığımız uygulamaları test edip çalıştırabilmemiz için bazı kurulumlara ihtiyacımız var. Bu nedenle öncelikle uygulama geliştirmeye başlamamız için tüm gerekli kurulum ve ayarları tam olarak yapmamız gerekiyor.

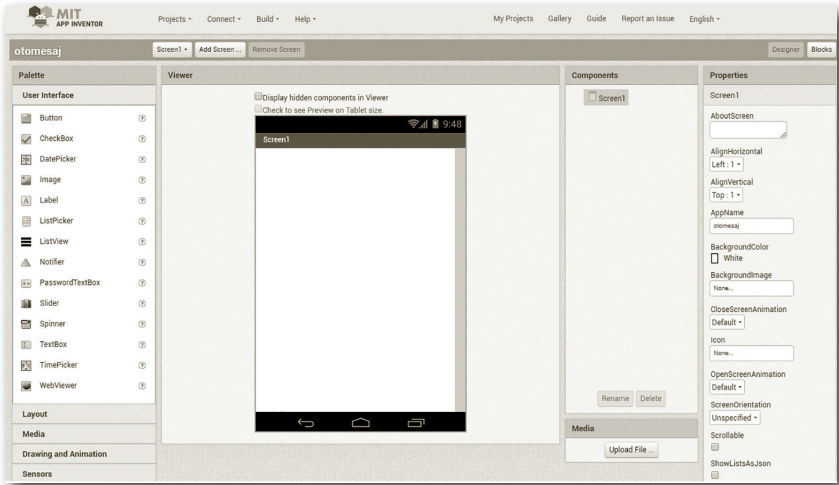
Hadi başlayalım!

Böylece dışımızı fırçalamış ve dış fırçalamak için gerekli olan algoritmayı da çıkarmış olduk. Hayatımızda yaptığımız her şey aslında birer algoritmadan ibarettir. Sabah kalkıp okula gitmemiz, yemek yememiz, eve geri gelmemiz, uyumamız gibi. Bilgisayar çalışma mantığında **Algoritma** çok ama çok önemlidir. Oynadığımız bilgisayar oyunları, programlar, web siteleri ve çoğu şey algoritması adım adım oluşturularak yapılmıştır.

## APP INVENTOR NEDİR?

MIT (*Massachusetts Institute of Technology*) tarafından geliştirilen **App Inventor** ile kolayca Android uygulamalar geliştirebiliriz. Sürükle bırak yöntemi ile çalışmaktadır. Oldukça eğlenceli olan bu platformda mobil uygulamalar geliştireceğiz ve Google Play Market'de yayınlamayı öğreneceğiz. App Inventor'ın IOS uygulamalar içinde çalışmalar yaptığı bilinmektedir. Belki de ilerleyen günlerde aynı yöntemler ile IOS uygulamalarda geliştirebiliriz.

App Inventor'a giriş için [www.appinventor.mit.edu](http://www.appinventor.mit.edu) adresini kullanacağız. Ancak App Inventor'a giriş yapmak için öncelikle **Gmail** hesabımız olması gerekmektedir.



App Inventor ile ilgili tanıtım videolarına; [youtube.com/user/mitappinventor](https://www.youtube.com/user/mitappinventor) adresinden erişebilirsiniz.

## GMAIL HESABI NASIL AÇILIR?

Gmail hesabı oluşturabilmek için öncelikle [www.accounts.google.com/SignUp](http://www.accounts.google.com/SignUp) adresine giriş yapmamız gerekmektedir.

**NOT**

Şifrenizin tahmin edilebilmesinin çok kolay olmamasına ve en az 8 karakterden oluşmasına dikkat ediniz.

Gerekli tüm bilgileri girdikten sonra İleri butonuna tıklayınız.

İleri butonuna tıkladıktan sonra karşımıza gelen sayfada da gerekli tüm bilgiler girildiğinde tekrar İleri butonuna tıklayınız.

**NOT**

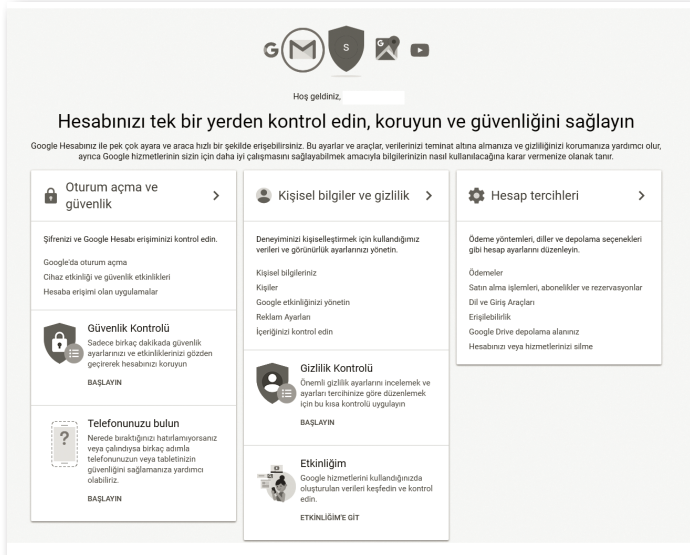
Telefon numarası bilgisi girmeniz hesabınızın şifresini unutmanız veya herhangi bir problemle karşılaşmanız durumunda, hesabınızı kurtarmanız için yardımcı olacaktır. Bu nedenle telefon bilgisi girmeniz önerilir. Ancak 18 yaşından küçükler bu işlemleri ebeveyn kontrolünde yapmalıdır.



Açılan gizlilik ve şartlar ekranını **Kabul Ediyorum** butonuna basarak geçiyoruz.

**NOT**

Eğer **Kabul ediyorum** butonunu göremezseniz sözleşmeyi tam olarak okuyunuz. Sözleşme tam olarak okunduğunda buton aktif hale gelecektir.

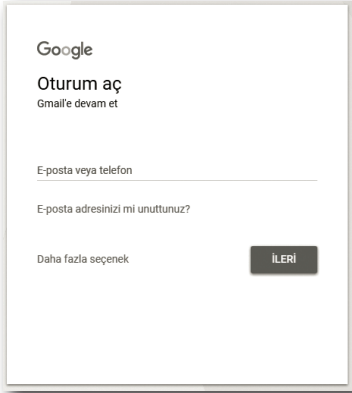




Bir önceki sayfadaki ekran ile karşılaştığınızda Gmail hesabınız başarı ile açılmış olacaktır. Yukarıdaki resimde bulunan Gmail ikonuna (daire içerisine alınmış alan) tıklayarak Gmail arayüzüne geçiş yapınız.

**App Inventor** uygulaması ile çalışmanız bittiğinde, bilgisayarınızdan kalkmadan önce **Gmail** hesabınızdan çıkış yapmayı unutmayınız. Hesaplarınızı çıkış yapmadan açık durumda bırakırsanız eğer, kötü niyetli insanlar tarafından kullanılabilir. Gmail hesabınızdan çıkış yapabilmek için, sağ üst köşede bulunan (yukarıdaki resimde koyu renkli daire) ikona tıklayınız. Açılan ekranda Oturumu kapat butonuna tıklayınız. Böylece Gmail hesabımızdan çıkış yapılmış olacaktır.

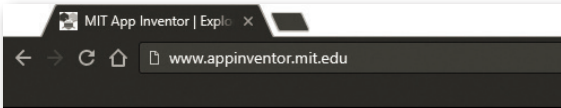
Tekrar Gmail hesabımıza erişmek için ise; [accounts.google.com](https://accounts.google.com) adresine giriş yapmalıyız.



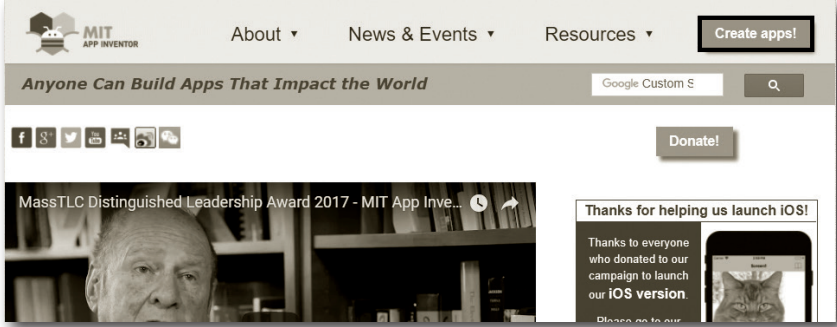
Açılan ekrana öncelikle oluşturduğumuz kullanıcı adınızı giriniz. Kullanıcı adınızı girdikten sonra **İleri** butonuna tıklayınız. Açılan ekranda şifrenizi giriniz ve tekrar **İleri** butonuna tıklayınız. Böylece Gmail hesabınıza giriş yapabilirsiniz.

## NASIL GİRİŞ YAPILIR?

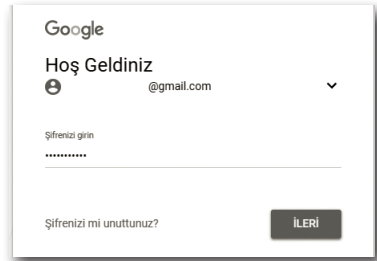
Herhangi bir tarayıcıyı kullanarak programa ulaşım sağlayabiliriz. Tek yapmamız gereken adres çubuğuna [appinventor.mit.edu](http://www.appinventor.mit.edu) yazmak olacaktır.



İkinci adımda ise karşımıza programın web sitesi geliyor. Sitenin sağ üst kısmında bulunan **Create apps!** butonuna tıklıyoruz. Bu işlemden sonra, eğer daha önce giriş yapmış isek, web sitesi bizi programın giriş sayfasına yönlendirecektir.

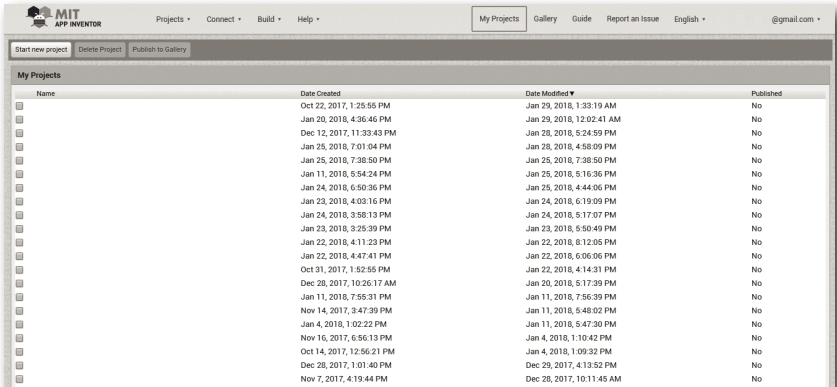


İlk defa giriş yapıyorsak, bir adet **gmail** hesabına ihtiyacımız olacak. Eğer hesabınız yoksa, bir önceki konumuzda anlatıldığı gibi kendi hesabınızı kolayca oluşturabilirsiniz. Oluşturduğunuz ya da var olan hesabınızı ve şifresini girerek sisteme giriş yapıyoruz.



Hesap girişi işlemlerini tamamladıktan sonra, web sitesi sizi programın giriş sayfasına yönlendirecektir.

**NOT** Gmail girişi sırasında eğer isterse; gerekli izinleri vermeyi unutmayınız.



# 3

## UYGULAMALAR

### BU BÖLÜMDE

Midillimi Seviyorum	44
Hayvan Seslerini Öğreniyorum	52
Piyanom	61
Konuşanları Yazıyorum	70
Havuçları Topluyorum	82
Bilgi Yarışması	93
Şu anda Ders Çalışıyorum	104
Çizim Programı Yapıyorum	111
Pedometre (Adımsayar) ile Kalori Hesaplama	118
Geri Dönüşüm Oyunu	123
WebViewler	128
Neler Öğrendik?	130

Bu bölümde yer alan uygulamalar şu şekilde:

- » Midillimi Seviyorum
- » Hayvan Seslerini Öğreniyorum
- » Piyanom
- » Konuşanları Yazıyorum
- » Havuçları Topluyorum
- » Bilgi Yarışması
- » Şu anda Ders Çalışıyorum
- » Çizim Programı Yapıyorum
- » Pedometre (Adımsayar) ile Kalori Hesaplama
- » Geri Dönüşüm Oyunu

**Midillimi Seviyorum - Algoritma**

1. Yeni proje oluştur.
2. Tasarıma başla.
3. Canvas bileşenini ekle.
4. Canvas'ın boyutlarını düzenle.
5. Canvas bileşenine arkaplan ekle.
6. Sound bileşenini ekle.
7. Sound bileşenine kaynak ekle.
8. Tasarımı bitir.
9. Kodlamaya başla
10. Ekranın üzerinde gezinme kodunu ekle (Canvas Dragged).
11. Ekranın üzerinde gezinildiğinde ses çalması için gerekli kodu ekle (Sound Play).
12. Kodlamayı bitir.
13. Programı çalıştır.

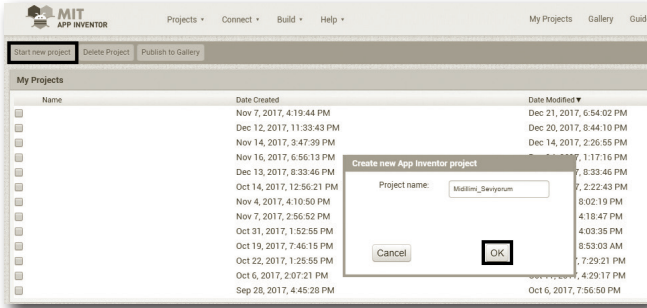
**NOT**

Algoritmayı kısaca işlem basamakları olarak tanımlayabiliriz. Örneğin sabah evden çıkıp okula gitmemize kadar gerçekleştirdiğimiz her adım bir algoritma komutudur.

**MİDİLLİMİ SEVİYORUM**

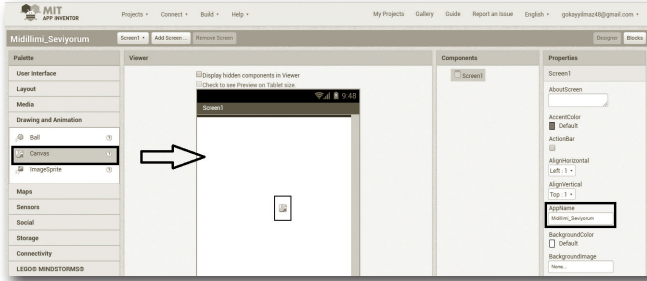
Bu programı yazmak için bir adet resim dosyasına ve bir adet ses dosyasına ihtiyacımız olacaktır. Bu dosyaları, ilgili bölümdeki linkleri kullanarak kolayca bilgisayarınıza indirebilir, derseniz farklı dosyalar da (doğru dosya uzantıları olmak şartıyla) kullanabilirsiniz.

İlk olarak **Start new project** butonu ile yeni projemizi oluşturmaya başlıyoruz. **Project name** için istediğiniz ismi verebilirsiniz; biz bu uygulamamızda **Midillimi\_Seviyorum** ismini kullanacağız. İsmi yazıp **OK** diyebiliriz.

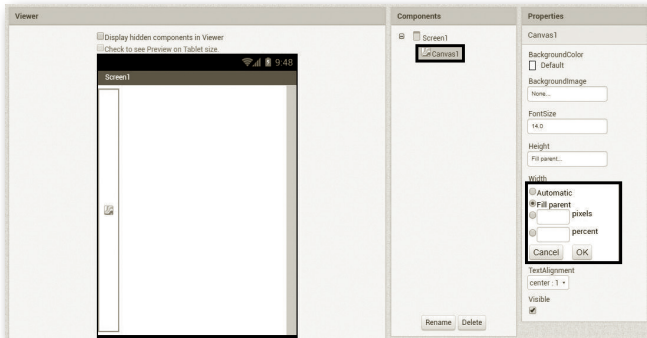


Sıradaki adımda ise uygulamamız için kullanacağımız bileşenleri ekleyeceğiz. Bu uygulamada iki adet bileşene ihtiyacımız olacak. Bunlar; Canvas ve Sound. Canvas bileşenini şekilde görüldüğü gibi sürükleyip; ekranımızın üstüne bırakıyoruz.

Dilersek resimde görülen AppName ile programımızın başlığını da değiştirebiliriz.

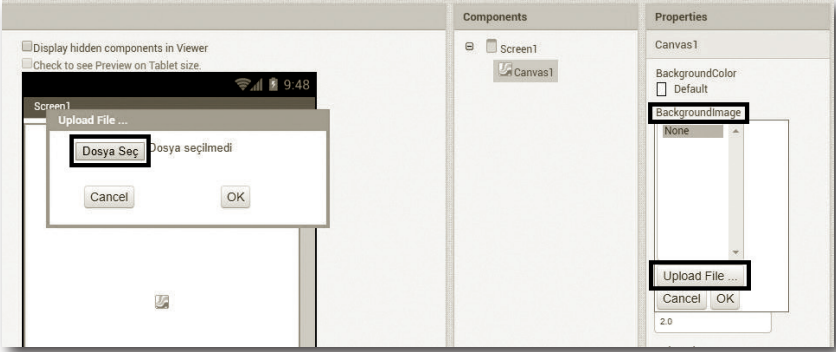


Eklediğimiz bileşen varsayılan olarak boyutu küçük olarak karşımıza gelecektir. Aşağıdaki şekilde görüldüğü gibi, Height ve Width değerlerini Fill parent olarak ayarlıyoruz. Bu özellik sayesinde bileşenimiz ekranımızı tam olarak kaplayacaktır.

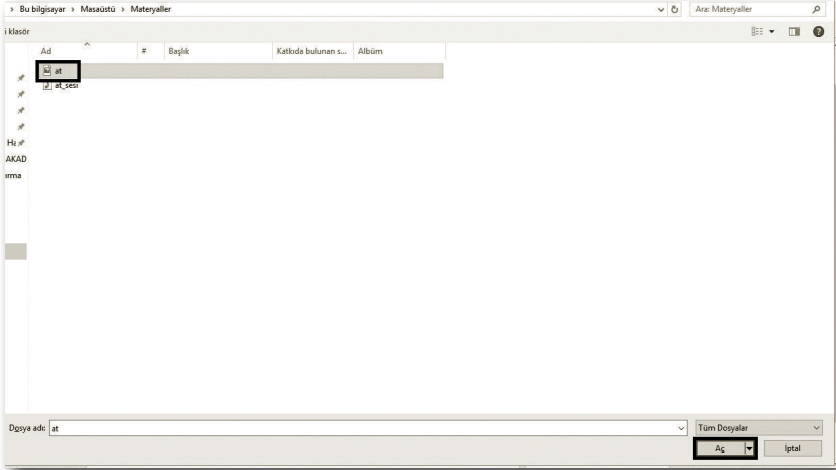


Canvas'ın boyutunu ayarladığımızı göre; artık resmimizi ekleyebiliriz.

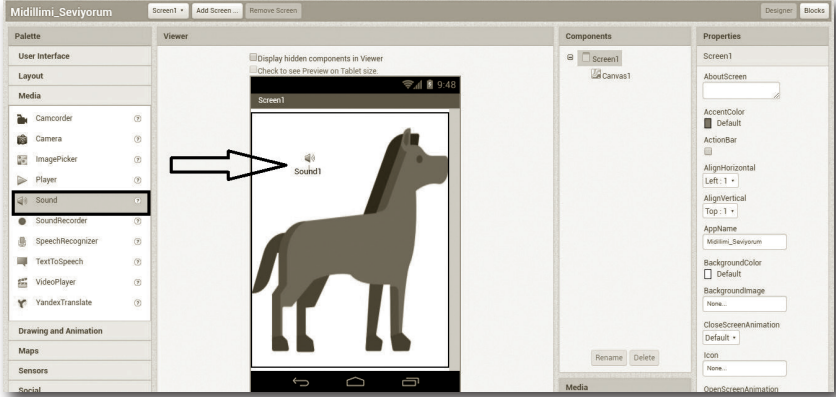
Canvas1 seçiliyken; sağ tarafta bulunan Properties sekmesi altında BackgroundImage için **Upload File** diyoruz ve açılan pencereden **Dosya Seç**'e tıklarız. (Aşağıdaki resim)



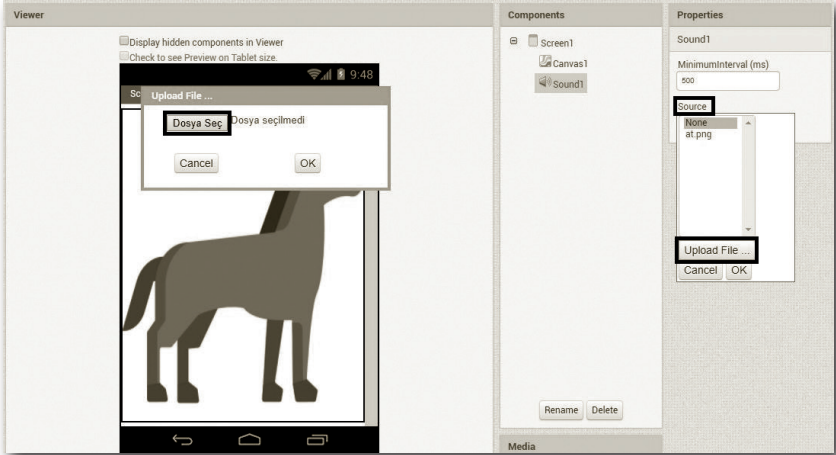
Hatırlayacağımız üzere; bir adet resim dosyasına ve bir adet ses dosyasına ihtiyacımız olduğunu söylemiştik. Bu adımda benim at olarak adlandırdığım resim dosyasına tıklarız ve **Aç** diyoruz. Ardından **OK** diyerek ekleme işlemini tamamlayabiliriz (bir sonraki resim).



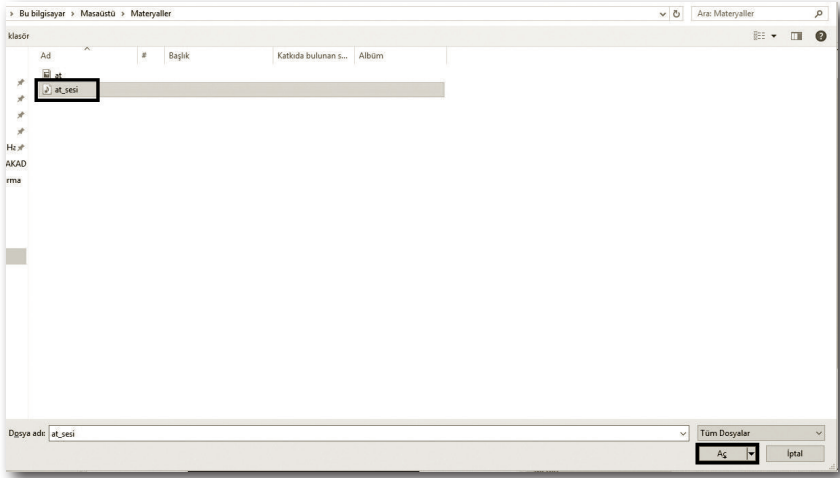
Resmimiz eklendikten sonra, sıradaki bileşen olan Sound bileşenini ekleyeceğiz. Canvas'ı eklerken kullandığımız yöntemle Media sekmesi altında Sound bileşenini tutup, uygulamamızın üstüne bırakıyoruz (Bir sonraki resim). Bu bileşen uygulamada gözükmeyen, fakat arkaplanda çalışan bir bileşendir. Bu sebeple sürükleyip bıraktıktan sonra sağ tarafta bulunan Component sekmesinin altında gözüktüğüne emin olmalısınız.



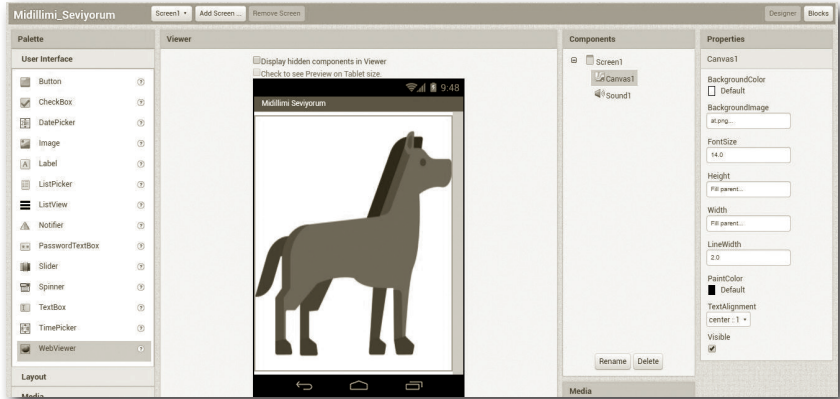
Canvas için yaptığımız Background Image seçme işleminin aynısını; Sound bileşenine Source eklemek için tekrarlıyoruz (aşağıdaki resim). Upload File butonuna, ardından Dosya Seç butonuna tıklıyoruz.



Aşağıdaki resimde görüldüğü gibi; benim at\_sesi olarak adlandırdığım ses dosyasını seçiyoruz ve Aç butonuna tıklıyoruz.

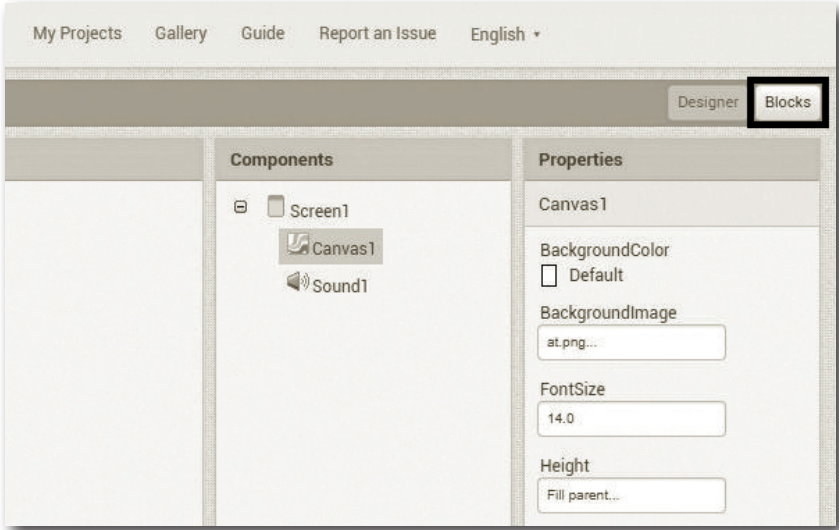


Tasarım ekranımızın son hali aşağıdaki gibidir.

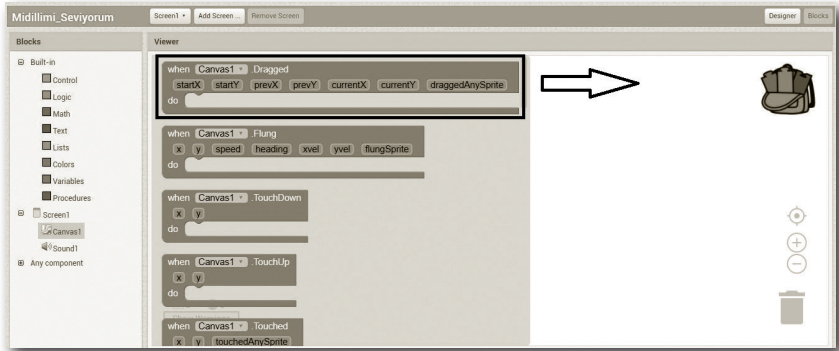


Uygulamamızın tasarımını bitirdiğimize göre; artık kodlama kısmına geçebiliriz. Programın sağ üst kısmında bulunan **Blocks** butonuyla aşağıdaki kodlama ekranına geçiyoruz.





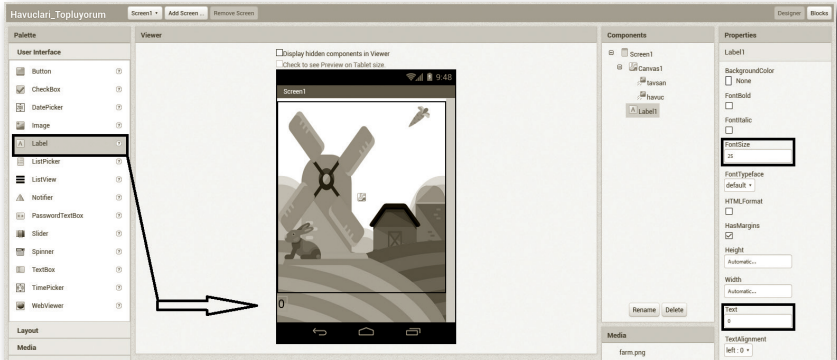
**Blocks** sekmesi altında, Canvas1 seçiliyken; when Canvas1. Dragged bloğunu, aşağıdaki resimde görüldüğü gibi blok ekranımıza sürükleyip bırakıyoruz:



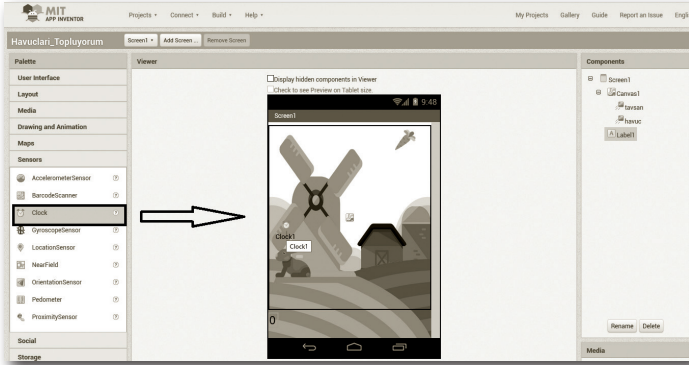
Daha sonra ise Sound1 bileşeni seçiliyken ca11 Sound1.Play' bloğunu, blok ekranımıza sürükleyip bırakıyoruz (aşağıdaki resim).



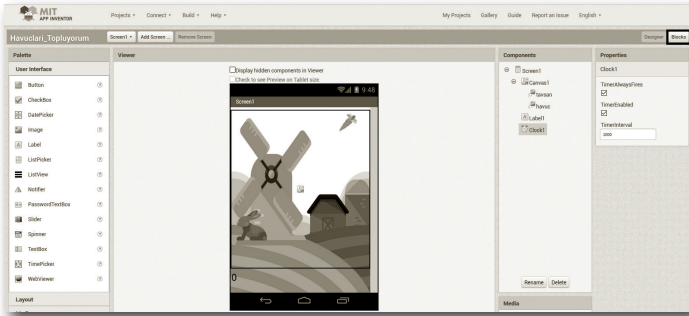
Canvas bileşenin boyunu ayarladıktan sonra, ekranın altında bıraktığımız boşluğa bir adet Label bileşeni ekliyoruz. Ekranın sağında bulunan Properties sekmesinin altında bulunan FontSize alanını 25 olarak, Text alanını ise 0 olarak değiştiriyoruz (aşağıdaki resim). Bu bileşeni skor tutmak için kullanacağız.



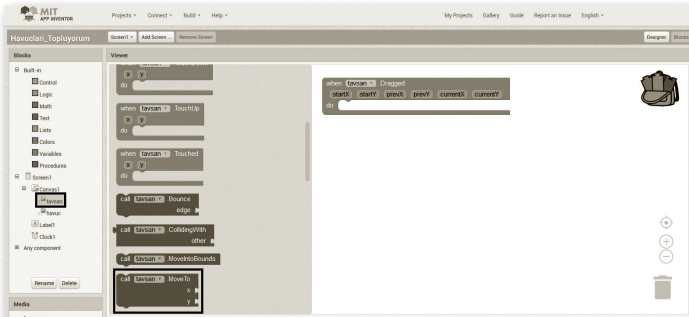
Tasarım ekranımızdaki son işlemimiz ise; havuc bileşenin yerini rastgele olarak değiştirmemize olanak sağlayacak Clock bileşenini eklemek olacak. Bunun için Sensors menüsünün altında bulunun Clock bileşenini şekilde görüldüğü gibi ekranımızın üstüne sürüklüyoruz. Bu bileşen non-visible, yani görünmeyen bileşenlerdendir. Varsayılan olarak her bir saniyede kendini tekrarlar. Bir saniye ara ile tekrarlamasını istediğimiz komutları, bu bileşeni kullanarak kontrol edebiliriz.



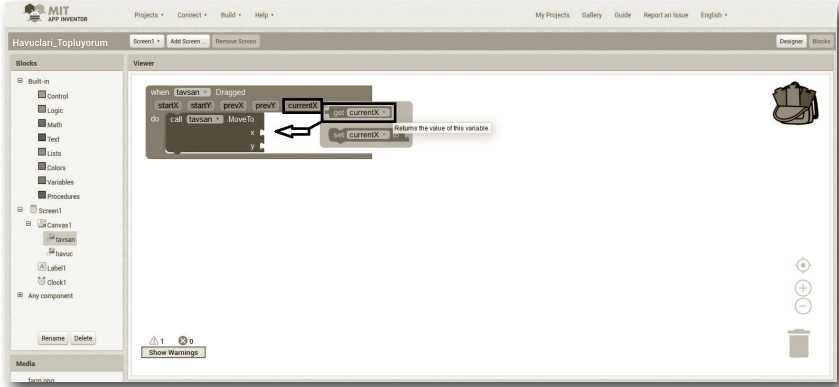
Tasarım ekranımızın son görüntüsü aşağıdaki gibidir. Sağ üstte bulunan **Blocks** butonuna tıklayarak, kodlama ekranına geçiş yapıyoruz.



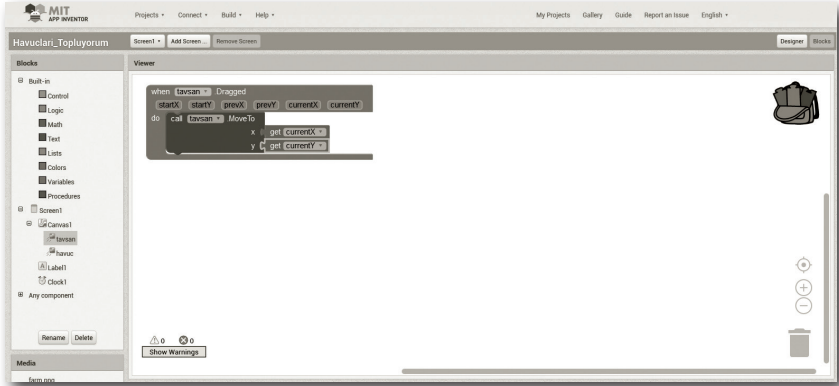
Kodlama işlemimize tavşan bileşenini sürükleyerek hareket ettirebilmemizi sağlayacak kod bloklarını düzenleyerek başlayacağız. Bu işlem için tavşan bloklarının altında bulunan **when tavşan.Dragged** komutunu kod alanımıza sürüklüyoruz. Daha sonra ise aynı bölümde bulunan **call1.tavşan.MoveTo** bloğunu sürükleyerek (bir sonraki resimdeki gibi), **when tavşan.Dragged** bloğunun içine yerleştiriyoruz.



Sıradaki işlemde ise sürüklenme işleminden sonra tavşanın taşınacağı  $x$  ve  $y$  noktalarını belirleyeceğiz. Bunun için  $currentX$  noktasını  $x$  bölümüne,  $currentY$  noktasını  $y$  bölümüne takıyoruz.



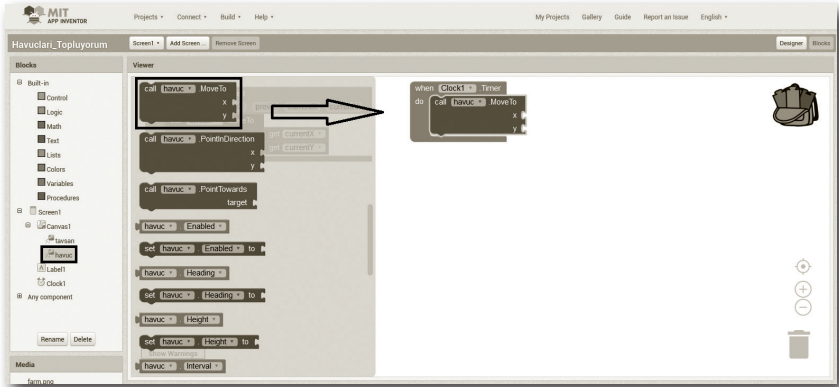
Bu işlem sonrasındaki görüntü aşağıdaki gibidir:



Bir sonraki işlemimiz ise; havuç bileşeninin belirli zaman aralığında ve rastgele olarak yer değiştirmesini sağlayacak kodlama işlemi gerçekleştirilmek. Bu durumu sağlamak için ilk olarak `Clock1` bloklarının altında bulunan `when Clock1`. Tımer bloğunu kod alanına sürüklüyoruz (aşağıdaki resim).



Clock bileşenini eklerken bahsettiğimiz gibi; her bir saniyede tekrarlayacak işlemleri bu kod bloğu ile kontrol edebiliriz. Her saniyede havuc bileşeninin yer değiştirmesini istediğimiz için when Clock1.Timer bloğunun içine call havuc.MoveTo bloğunu yerleştiriyoruz (aşağıdaki gibi).



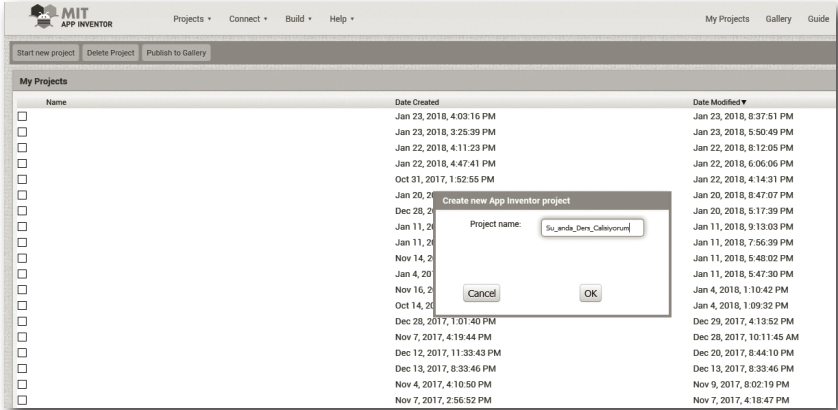
Şimdi ise havuc bileşeni her saniyede taşıyacağımız x ve y noktasını belirteceğiz. Bu noktaların her seferinde farklı olması için, Math bloklarının altında bulunan random integer from 1 to 100 bloğunu, bir sonraki resimde görüldüğü gibi yerleştiriyoruz ve içeriğini de resimdeki gibi değiştiriyoruz. Bu blok sayesinde program, belirttiğimiz sayı aralığında rastgele sayılar üretecek ve havuc bileşeni rastgele olarak yer değiştirmiş olacaktır.

8. Tasarımı bitir.
9. Kodlamaya başla.
10. Arama bittiğinde bloğunu sürükle (when PhoneCall1.PhoneCallEnded).
11. Eğer status değişkeni 1 ise → Texting bileşeninin telefon numarasını ayarla.  
→ Texting bileşeninin mesaj metnini ayarla.  
→ Mesajı gönder.
12. Kodlamayı bitir.
13. Programı çalıştır.

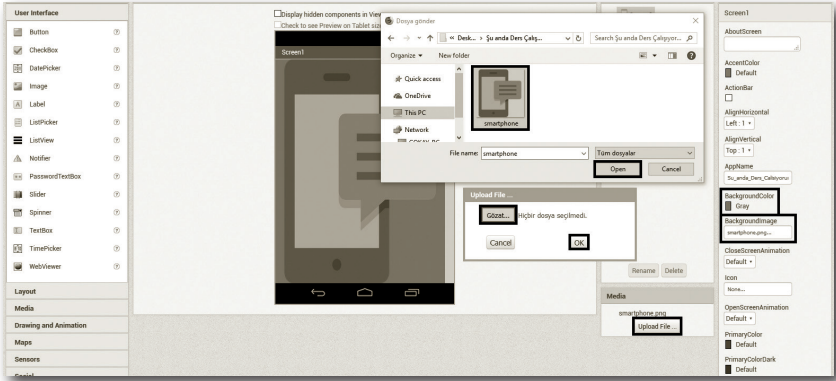
## ŞU ANDA DERS ÇALIŞIYORUM

Bu programı yazmak için bir, adet resim dosyasına ihtiyacımız olacaktır. Bu dosyayı, ilgili bölümdeki linkleri kullanarak kolayca bilgisayarınıza indirebilir, dilerse farklı dosyalar da (doğru dosya uzantıları olmak şartıyla) kullanabilirsiniz.

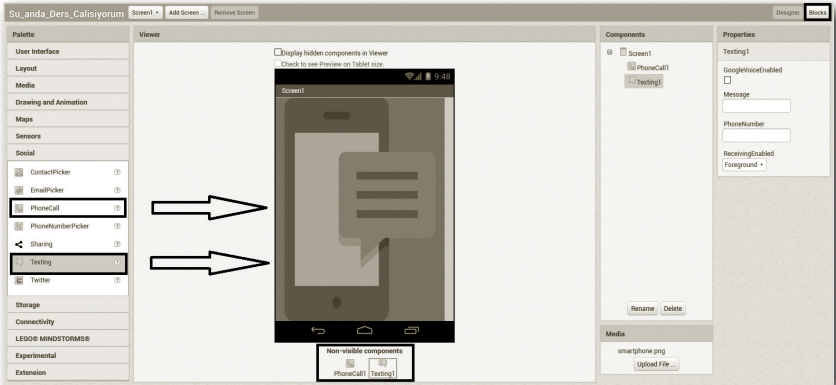
İlk olarak **Start new project** butonu ile yeni projemizi oluşturmaya başlıyoruz. **Project name** için istediğiniz ismi verebilirsiniz; biz bu uygulamamızda **Su\_anda\_Ders\_Calisiyorum** ismini kullanacağız. İsmi yazıp **OK** diyebiliriz.



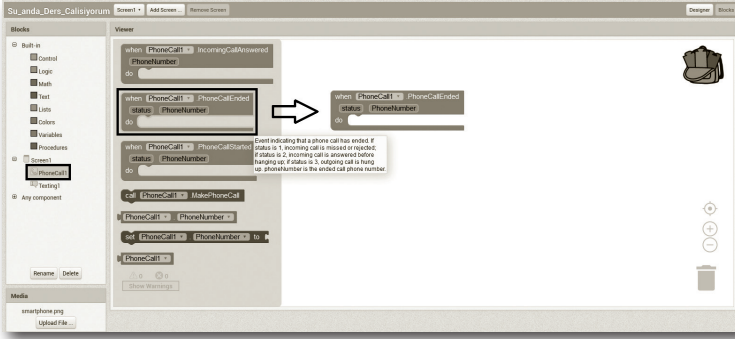
Tasarım işlemlerimize, arkaplan olarak kullanacağımız resmi ekleyerek başlıyoruz. **Upload File** butonuna basarak, indirdiğimiz ya da kullanmak istediğimiz resmi seçiyoruz ve **OK** diyerek sunucuya yüklüyoruz. Daha sonra ise yüklediğimiz bu resmi arkaplan olarak ayarlamak için; **BackgroundImage** alanına tıklayarak, resmimizi seçiyoruz. **BackgroundColor** olarak **Gray** rengini seçiyoruz. Bu seçimleri isteğiniz doğrultusunda değiştirebilirsiniz.



Sıradaki işlemimize kullanacağımız bileşenleri ekleyerek devam ediyoruz. Bu program için iki adet bileşene ihtiyacımız olacak; **PhoneCall** ve **Texting**. İki bileşeni de şekilde görüldüğü gibi ekranımızın üstüne sürüklüyoruz. Daha sonra ise; ekranın sağ üstünde bulunan **Blocks** butonu ile kodlama alanımıza geçiyoruz.



Kodlama işlemimize PhoneCall11 bloklarının altında bulunan when PhoneCall11. PhoneCall11Ended bloğunu kodlama alanına sürükleyerek başlıyoruz. Bu blok sayesinde gelen aramaları kontrol edebileceğiz.



Sıradaki işlemimize **Control** bloklarının altında bulunan **if** bloğunu aşağıda görüldüğü gibi yerleştirerek devam ediyoruz. Bu blok, aldığımız bölümün isminden de anlaşılacağı gibi kontrol işlemleri yapmamıza olanak sağlar.



Bu örneğimizde iki değişkeni karşılaştırarak; eğer bu değerler birbirine eşit ise gerekli işlemleri yaptıracağız. Bu sebeple **Logic** bloklarının altında bulunan ve eşitlik karşılaştırması yapmamıza olanak sağlayan bloğu **if** bloğuna takıyoruz.

