



T.C.

MARMARA ÜNİVERSİTESİ

TEKNİK BİLİMLER MESLEK YÜKSEKOKULU

BİLGİSAYAR TEKNOLOJİLERİ BÖLÜMÜ

BİLGİSAYAR PROGRAMCILIĞI PROGRAMI

Marmara Drone

AR. Drone Aracını Bilgisayar ve Android Mobil Sistemden Kontrolü

360111033 - Fırat BİŞKİN

360111006 - Mehmet SABANCIOĞLU

Danışman: Doç. Dr. Vedat TOPUZ

İSTANBUL 2013



T.C.

MARMARA ÜNİVERSİTESİ

TEKNİK BİLİMLER MESLEK YÜKSEKOKULU

BİLGİSAYAR TEKNOLOJİLERİ BÖLÜMÜ

BİLGİSAYAR PROGRAMCILIĞI PROGRAMI

Marmara Drone

AR. Drone Aracını Bilgisayar ve Android Mobil Sistemden Kontrolü

360111033 - Fırat BİŞKİN

360111006 - Mehmet SABANCIOĞLU

Juri Üyeleri

Doç. Dr. Vedat TOPUZ Öğr.Gör. Dr. Aysun ALTIKARDEŞ Öğr.Gör. Murat TOPAL

İSTANBUL 2013

MARMARA DRONE



ÖNSÖZ

Java ve Android tabanında geliştirilmek amacıyla AR. Drone 2.0'lı 4 motorlu, kablosuz kontrollü hava aracı üzerinde yazılım geliştirme hedeflenildi. Toplam proje bütçesi 950 TL tutmuştur. Projenin Masaüstü uygulaması JAVA tabanlı olup, bilgisayar üzerinden kontrol amacına ulaşılmıştır.

Bunun yanında Android tabanında cihazı kontrol etmek amacıyla yazılmış olan uygulama da platformdan kaynaklanan sorunlardan ötürü uygulama da bir takım çakışmalar olmuştur. Bu çakışmalar sonucu android uygulaması üzerinden cihaza erişim mümkün kılınmamıştır. Bu yüzden Projenin ikinci aşaması olan Android üzerinden kontrol süre ve bir takım diğer etkenlerden dolayı başarısız olmuştur. Mevcut Android Proje, sorun yaşanan yere kadar proje dahil edilmiştir.

Bu proje sayesinde başta Havacılığa Olan merakımızı gidermenin yanında, hava araçları hakkında mantıksal yordamlı sonuçlar çıkarılmış olup. Ayrıca Java, Android ve Eclipse üçlüsü ile uygulama geliştirme bilgisi kazanılmıştır.



İçindekiler Tablosu

AR. Drone Nedir? _____	1
Masaüstü Uygulaması Ön İzleme _____	2
Yönetimsel Fonksiyonlar _____	3-8
Android Mobil Sistem Ön İzleme _____	9
Main.JAVA _____	10-15
MarmaraDroneApplication.JAVA _____	16
Java ve Android Platformlarının Kurulumu _____	17-27
Kaynakça _____	28
Projeden Resimler _____	29
Kapanış _____	30



AR. Drone Nedir?

Parrot'ın uçan robotu Ar. Drone yapabildikleriyle sizi şaşırtacak. Android ve IOS işletim sistemli Mobil cihazlardan kontrol edilebilen Ar. Drone, içindeki mini bilgisayar sayesinde rüzgâra karşı konum koruma, yükseklik ölçme ve otomatik iniş gibi birçok özelliikle donatılmış. Aynı Zamanda kameralarından elde ettiği görüntüleri de sürekli ekranınıza aktarabiliyor!

Parrot AR Drone 2.0 Özellikleri

- Jenerasyon AR. Drone 'un zayıf noktaları tespit edilerek kuvvetlendirilmiştir. Ön HD kamera kuvvetlendirilmiş kapalı mekân kabini ve orta çapraz sayesinde tam koruma altındadır.
- AR. Drone 2,0'ın tabanındaki elektronik parçalar su geçirmez film ile kapladı. Bu sayede AR. Drone 2.0 suya inebilir. AR. Drone 2.0'ı karda veya çimli bir sabahta sorunsuz uçurabilirsiniz.
- İlk jenerasyon AR. Drone 'da 6 metre ile sınırlı otomatik stabilizasyon özelliği, AR. Drone 2.0 ile tarihe karışıyor. Otomatik stabilizasyon artık her yükseklikte mevcut.
- Parrot patentli AR. Drone 2.0'a özgü tam kontrol pilot modu. Bu özel pilotluk modu sayesinde kullanıcı, AR. Drone 'un duruş yönüne aldırmadan telefonunu hangi yöne yatırır o yöne doğru hareket ettirebilecek.
- AR. Dreeflight 2.0: Parrot AR. Drone için pilotluk uygulaması; Wi-Fi üzerinden canlı görüntü aktarımı sayesinde uzaktan kumanda.
- AR. Pursuit: 2 kişilik AR. Drone oyunu; sanal makineli tüfekler ve roketler ile AR. Drone'u avlayın.
- AR. Race; AR. Dron yarış uygulaması. Kullanıcı zamana karşı ya da diğer AR. Drone kullanıcılarına karşı yarışır.
- WiFi kontrol aralığı: 50m
- Ana Kamera HD 1280x720p, geniş açı lens ile
- Otomatik iniş: Pilin azalması ile otomatik olarak kendisi iniş yapar
- Pil: 3 hücre 11.1V 30w 1000 mAh lityum polimer 14g paketi
- Uçuş süresi: 12 dk
- Motorlar: 4,000 rpm 4x 14.5w fırçasız elektrik motorla



AR.Drone Bağlantısı

```
private KomutYoneticisi manager = null;  
public boolean connect()  
{  
    if(inetaddr==null)  
    {  
        inetaddr=getInetAddress(ipaddr);  
    }  
    manager=new KomutYoneticisi(inetaddr);  
    return manager.connect(ARDroneUtils.PORT);  
}
```

Video Bağlantısı

```
public boolean connectVideo()  
{  
    if(inetaddr==null)  
    {  
        inetaddr=getInetAddress(ipaddr);  
    }  
    videoManager=new VideoManager(inetaddr, manager);  
    videoManager.setImageListener(new ImageListener() {  
        public void imageUpdated(BufferedImage image)  
        {  
            if(imageListener!=null)  
            {  
                imageListener.imageUpdated(image);  
            }  
        }  
    });  
    return videoManager.connect(ARDroneUtils.VIDEO_PORT);  
}
```


NavData Bağlantısı



```

public boolean connectNav()
{
    if(inetaddr==null)
    {
        inetaddr=getInetAddress(ipaddr);
    }
    navdataManager=new NavDataManager(inetaddr, manager);
    navdataManager.setAttitudeListener(new YukseklikAlgilayici()
    {
        @Override
        public void attitudeUpdated(float pitch, float roll, float yaw, int altitude)
        {
            if(attitudeListener!=null)
            {
                attitudeListener.attitudeUpdated(pitch, roll, yaw, altitude);
            }
        }
    });
    navdataManager.setBatteryListener(new BataryaAlgilayici()
    {
        @Override
        public void batteryLevelChanged(int percentage)
        {
            if(batteryListener!=null)
            {
                batteryListener.batteryLevelChanged(percentage);
            }
        }
    });
    navdataManager.setStateListener(new KonumAlgilayici()
    {
        @Override
        public void stateChanged(DroneKonumu state)
        {
            if(stateListener!=null)
            {
                stateListener.stateChanged(state);
            }
        }
    });
}

```




```
navdataManager.setVelocityListener(new HizAlgilyici() {
    @Override
    public void velocityChanged(float vx, float vy, float vz)
    {
        if(velocityListener!=null)
        {
            velocityListener.velocityChanged(vx, vy, vz);
        }
    }
});
navdataManager.setNavDataListener(new NavDataListener()
{
    @Override
    public void navDataUpdated(NavData navData)
    {
        if(navDataListener!=null)
        {
            navDataListener.navDataUpdated(navData);
        }
    }
});

return navdataManager.connect(ARDroneUtils.NAV_PORT);
}
```

Başlatma Fonksiyonu

```
public void start()
{
    if(manager!=null)
        new Thread(manager).start();
    if(videoManager!=null)
        new Thread(videoManager).start();
    if(navdataManager!=null)
        new Thread(navdataManager).start();
}
```



Komutlar

```
@Override
    public void takeOff() {
        if(manager!=null)
            manager.takeOff();
    }

@Override
    public void landing() {
        if(manager!=null)
            manager.landing();
    }

@Override
    public void reset() {
        if(manager!=null)
            manager.reset();
    }

@Override
    public void forward() {
        if(manager!=null)
            manager.forward();
    }

@Override
    public void forward(int speed) {
        if(manager!=null)
            manager.forward(speed);
    }

@Override
    public void backward() {
        if(manager!=null)
            manager.backward();
    }

@Override
    public void backward(int speed) {
        if(manager!=null)
            manager.backward(speed);
    }
```



```
@Override
    public void spinRight() {
        if(manager!=null)
            manager.spinRight();
    }

@Override
    public void spinRight(int speed) {
        if(manager!=null)
            manager.spinRight(speed);
    }

@Override
    public void spinLeft() {
        if(manager!=null)
            manager.spinLeft();
    }

@Override
    public void spinLeft(int speed) {
        if(manager!=null)
            manager.spinLeft(speed);
    }

@Override
    public void up() {
        if(manager!=null)
            manager.up();
    }

@Override
    public void up(int speed) {
        if(manager!=null)
            manager.up(speed);
    }

@Override
    public void down() {
        if(manager!=null)
            manager.down();
    }
```



```
@Override
    public void down(int speed) {
        if(manager!=null)
            manager.down(speed);
    }

@Override
    public void goRight() {
        if(manager!=null)
            manager.goRight();
    }

@Override
    public void goRight(int speed) {
        if(manager!=null)
            manager.goRight(speed);
    }

@Override
    public void goLeft() {
        if(manager!=null)
            manager.goLeft();
    }

@Override
    public void goLeft(int speed) {
        if(manager!=null)
            manager.goLeft(speed);
    }

@Override
    public void setSpeed(int speed) {
        if(manager!=null)
            manager.setSpeed(speed);
    }

@Override
    public void stop() {
        if(manager!=null)
            manager.stop();
    }
```



Android Mobil Sistem Ön İzleme





Main.JAVA

Mobil uygulamanın main bloğu, program burda çalışır. Button fonksiyonları, yönetimsel fonksiyonlar burdadır.

```
package com.flbilisim.marmaradrone;
import android.app.Activity;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.net.wifi.WifiManager;
import android.os.Bundle;
import android.view.KeyEvent;
import android.view.Menu;
import android.view.MotionEvent;
import android.view.View;
import android.view.View.OnClickListener;
import android.view.View.OnTouchListener;
import android.widget.Button;
import android.widget.TextView;
import com.shigeodayo.ardrone.ARDrone;
import com.flbilisim.marmaradrone.R;
import com.flbilisim.marmaradrone.MarmaraDroneApplication;
```

```
public class Main extends Activity
{
    @Override
    protected void onCreate(Bundle savedInstanceState)
    {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        initButtons(); //Button fonk. [Çakışmanın olduğu yer]
    }
}
```

```
private void initButtons()
{
    MarmaraDroneApplication app = (MarmaraDroneApplication)getApplication();
    final ARDrone drone = app.getARDrone();
    Button forward = (Button)findViewById(R.id.cmd_forward);
    forward.setOnTouchListener(new OnTouchListener()
    {
```




```

        public boolean onTouch(View v, MotionEvent event)
        {
            if(event.getAction() == MotionEvent.ACTION_DOWN)
                drone.forward(20);
            else if (event.getAction() == MotionEvent.ACTION_UP)
                drone.stop();
            return true;
        }
    });

    Button connect = (Button)findViewById(R.id.button1);
    connect.setOnClickListener(new OnClickListener()
    {
        public void onClick(View v)
        {
            WifiManager wifi = (WifiManager) getSystemService(Context.WIFI_SERVICE);
            final TextView text = (TextView)findViewById(R.id.text_init);
            text.append("\nBağlanildi " + wifi.getConnectionInfo().getSSID());

            try
            {
                text.append("\n\n Drone Kontrolleri Bağlandı \n");
                drone.connect();
                text.append(" Drone Navdata Socket Bağlandı\n");
                drone.connectNav();
                // text.append(" Drone Video bağlandı\n");
                // ardrone.connectVideo();
                text.append("Başlat Drone\n");
                drone.start();

                text.append("Maximim Yukseklik 5 Metre(15ft)\n\n");
                drone.setMaxAltitude(5000);
                // maximum 5 metre
            }
            catch(Exception exc)
            {
                exc.printStackTrace();

                if (drone != null)
                    drone.disconnect();
            }
        }
    });

```




```
Button backward = (Button)findViewById(R.id.cmd_backward);
backward.setOnClickListener(new OnTouchListener()
```

```
{
    public boolean onTouch(View v, MotionEvent event)
    {
        if(event.getAction() == MotionEvent.ACTION_DOWN)
            drone.backward(20);
        else if (event.getAction() == MotionEvent.ACTION_UP)
            drone.stop();
        return true;
    }
});
```

```
Button left = (Button)findViewById(R.id.cmd_left);
left.setOnClickListener(new OnTouchListener()
```

```
{
    public boolean onTouch(View v, MotionEvent event)
    {
        if(event.getAction() == MotionEvent.ACTION_DOWN)
            drone.goLeft(20);
        else if (event.getAction() == MotionEvent.ACTION_UP)
            drone.stop();
        return true;
    }
});
```

```
Button right = (Button)findViewById(R.id.cmd_right);
right.setOnClickListener(new OnTouchListener()
```

```
{
    public boolean onTouch(View v, MotionEvent event)
    {
        if(event.getAction() == MotionEvent.ACTION_DOWN)
            drone.goRight(20);
        else if (event.getAction() == MotionEvent.ACTION_UP)
            drone.stop();
        return true;
    }
});
```



```

Button up = (Button)findViewById(R.id.cmd_up);
up.setOnClickListener(new OnTouchListener()
{
    public boolean onTouch(View v, MotionEvent event)
    {
        if(event.getAction() == MotionEvent.ACTION_DOWN)
            drone.up(40);
        else if (event.getAction() == MotionEvent.ACTION_UP)
            drone.stop();
        return true;
    }
});

Button down = (Button)findViewById(R.id.cmd_down);
down.setOnClickListener(new OnTouchListener()
{
    public boolean onTouch(View v, MotionEvent event)
    {
        if(event.getAction() == MotionEvent.ACTION_DOWN)
            drone.down(40);
        else if (event.getAction() == MotionEvent.ACTION_UP)
            drone.stop();
        return true;
    }
});

Button spinLeft = (Button)findViewById(R.id.cmd_spin_left);
spinLeft.setOnClickListener(new OnTouchListener()
{
    public boolean onTouch(View v, MotionEvent event)
    {
        if(event.getAction() == MotionEvent.ACTION_DOWN)
            drone.spinLeft(20);
        else if (event.getAction() == MotionEvent.ACTION_UP)
            drone.stop();
        return true;
    }
});

```



```

Button spinRight = (Button)findViewById(R.id.cmd_spin_right);
spinRight.setOnTouchListener(new OnTouchListener()
{
    public boolean onTouch(View v, MotionEvent event)
    {
        if(event.getAction() == MotionEvent.ACTION_DOWN)
            drone.spinRight(20);
        else if (event.getAction() == MotionEvent.ACTION_UP)
            drone.stop();
        return true;
    }
});

Button landing = (Button)findViewById(R.id.cmd_landing);
landing.setOnClickListener(new OnClickListener()
{
    public void onClick(View v)
    {
        drone.landing();
    }
});

Button emergency = (Button)findViewById(R.id.cmd_emergency);
emergency.setOnClickListener(new OnClickListener()
{
    public void onClick(View v)
    {
        drone.reset();
    }
});

public boolean onKeyDown(int keyCode, KeyEvent event)
{
    if (keyCode == KeyEvent.KEYCODE_BACK && event.getRepeatCount() == 0)
    {
        new AlertDialog.Builder(this).setMessage("Eğer çıkarsanız Drone bağlantısı kesilir !").setTitle("MarmaraDrone çıkmak istediğinize emin misiniz ?").setCancelable(false).setPositiveButton(android.R.string.ok, new DialogInterface.OnClickListener()
        {

```



```

        public void onClick(DialogInterface dialog, int whichButton)
        {
            MarmaraDroneApplication app =
            (MarmaraDroneApplication)getApplication();
            ARDrone drone = app.getARDrone();
            drone.disconnect();
            finish();
        }
    })
    .setNegativeButton(android.R.string.cancel, new DialogInterface.OnClickListener()
    {
        public void onClick(DialogInterface dialog, int whichButton)
        {
        }
    })
    .show();

    return true;
}

return super.onKeyDown(keyCode, event);
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.main, menu);
    return true;
}
}

```



MarmaraDroneApplication.JAVA

Drone bağlantısını kurar ve drone'u geri döndürür.

```
package com.flbilisim.marmaradrone;

import android.app.Application;
import com.shigeodayo.ardrone.ARDrone;

public class MarmaraDroneApplication extends Application
{
    private ARDrone drone;

    public void onCreate()
    {
        drone = new ARDrone("192.168.1.1");
    }

    public ARDrone getARDrone()
    {
        return drone;
    }
}
```

Java ve Android Platformlarının Kurulumu

- 1) Android java tabanlı olduğu için ilk olarak Java JDK(Java Development Kit)'yi kurulacak = [Java Platform \(JDK\) 7u3.](#)
- 2) Sonra Java JRE'yi kurulacak = [Java \(JRE\) 7u3.](#)
- 3) İde olarak Eclipse kullanıyoruz = [Eclipse IDE for Java Developers](#)
- 4) Daha Sonra Android Geliştirici Kiti kurulacak = [Android SDK.](#)
- 5) Son olarakda Eclipse için ADT(android development tools) kurulacak, Eclipse ile android projeleri oluşturabilmemize yarayan Eclipse eklentisidir.



eclipse-java-indigo-SR2-win32
132.094 KB



installer_r16-windows
Google Inc.



jdk-7u3-windows-i586
Java(TM) Platform SE binary
Oracle Corporation



jre-7u3-windows-i586
Java(TM) Platform SE binary
Oracle Corporation

Gerekli dosyalarımızı temin ettikten sonra. İlk olarak 1. ve 2. sıradaki Java JDK ve JRE'nin kurulumunu yapılacak. Java SDK'yı kurarken JRE'yide kurabiliyor, yükleme ekranında bunu görebiliriz. Eğer kurmazsa ardından JRE kurulması gerekir.

Here are the Java SE downloads in detail:

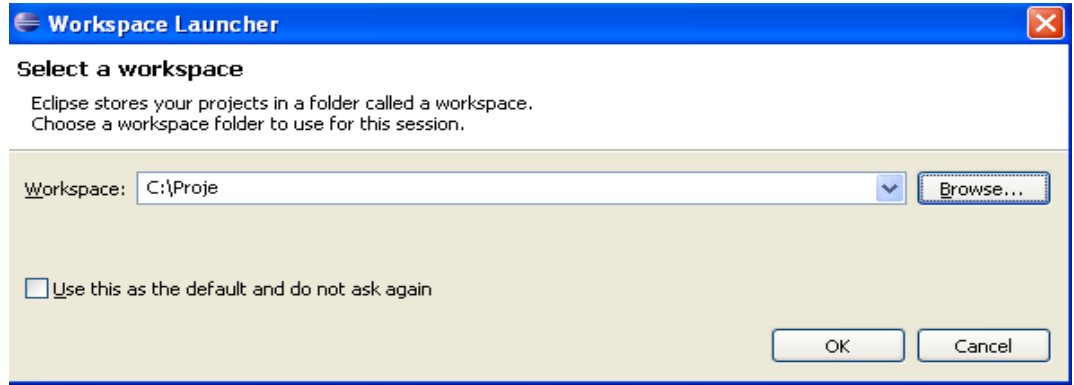
Java Platform, Standard Edition		
Java SE 7u3	JDK	JRE
This release includes security fixes. Learn more ▶	Download	Download
"What Java Do I Need?" You must have a copy of the JRE (Java Runtime Environment) on your system to run Java applications and applets. To develop Java applications and applets, you need the JDK (Java Development Kit), which includes the JRE.	JDK 7 Docs	JRE 7 Docs
	<ul style="list-style-type: none"> ▪ Installation Instructions ▪ ReadMe ▪ ReleaseNotes ▪ Oracle License ▪ Java SE Products ▪ Third Party Licenses ▪ Certified System Configurations 	<ul style="list-style-type: none"> ▪ Installation Instructions ▪ ReadMe ▪ ReleaseNotes ▪ Oracle License ▪ Java SE Products ▪ Third Party Licenses ▪ Certified System Configurations



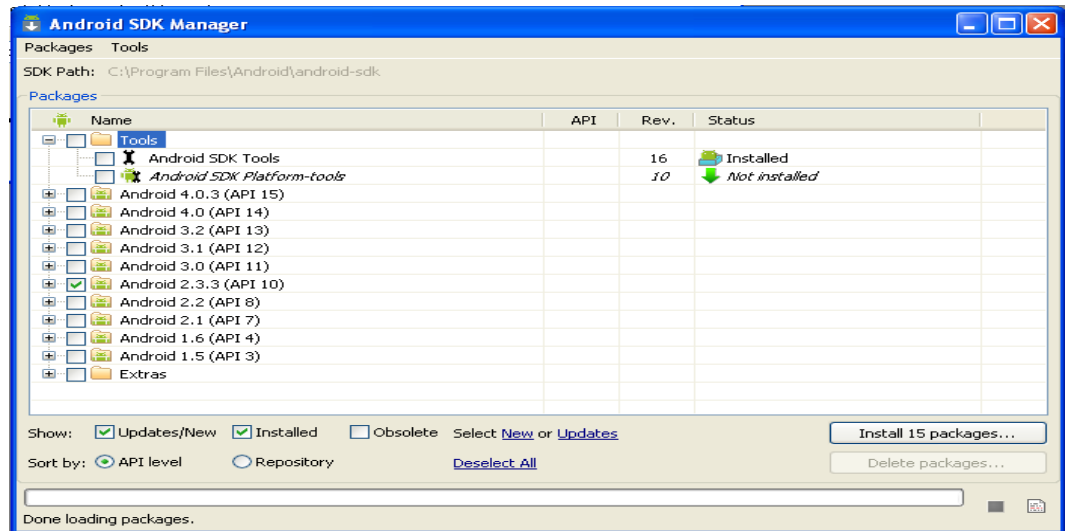
Daha sonra indirilen [Eclipse IDE for Java Developers](#) programını kuruyoruz. Kurmaktan kasıt Eclipse indirdiğinde zaten bir zip dosyası halinde geliyor. Zip dosyasını istenilen bir yere çıkartılması yeterlidir.



JDK'da kurulu olduğundan Eclipse çalışacaktır. Eclipse'i çalıştırılır ve çalışma klasörü sorusu ekranı görülür. Document And Settings klasörü yazılırken arada boşluklar olduğundan bazı ide'lerde sorunlar yaşanabiliyor. O yüzden klasörü direk C:'de tanımlamakta fayda var. Eclipse her açıldığında bu ekranla karşılaşmak istemiyor iseniz Use This as diye başlayan yerin seçilmesi yeterlidir.

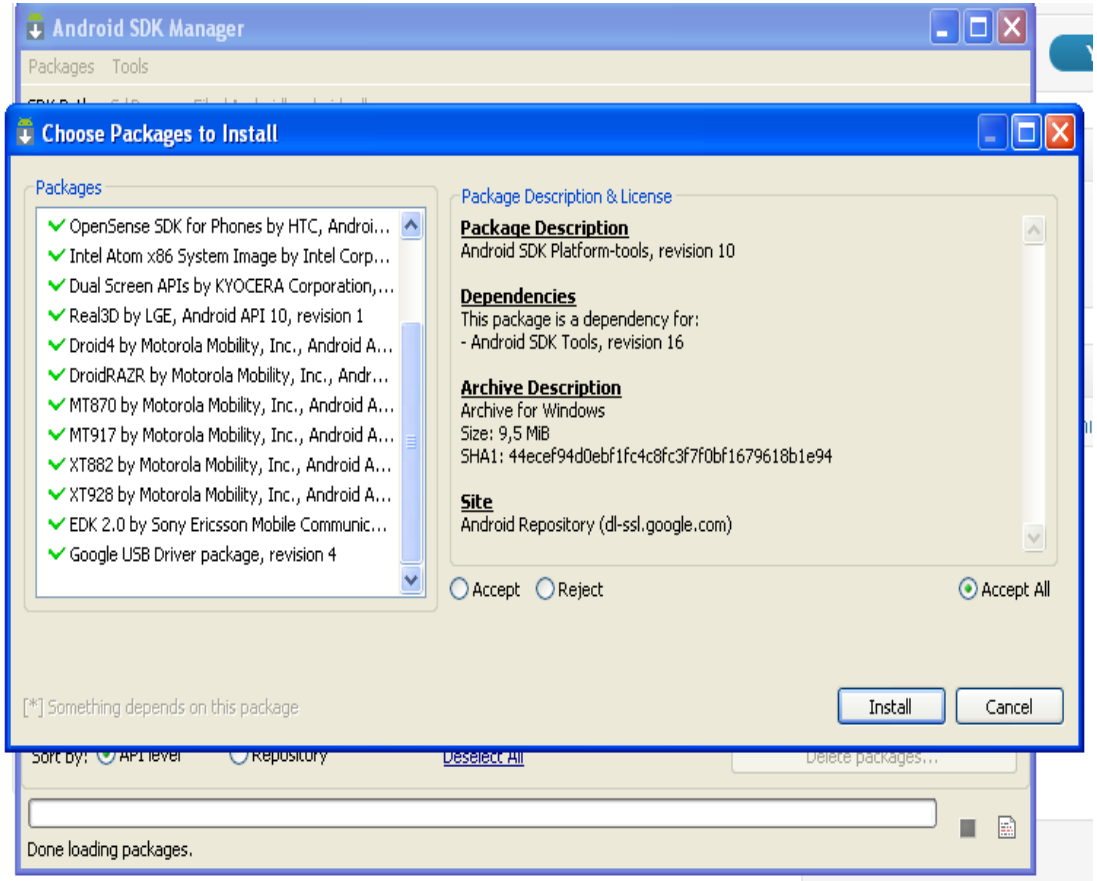


Android için indirmiş olan sdk'yı kurulur. Daha sonra Android SDK manager'ı açarak hangi Android platformu için program yazacak isek onun API'sini indirilir. Proje için 2.3.3 ve üstü sürümü seçilebilir.



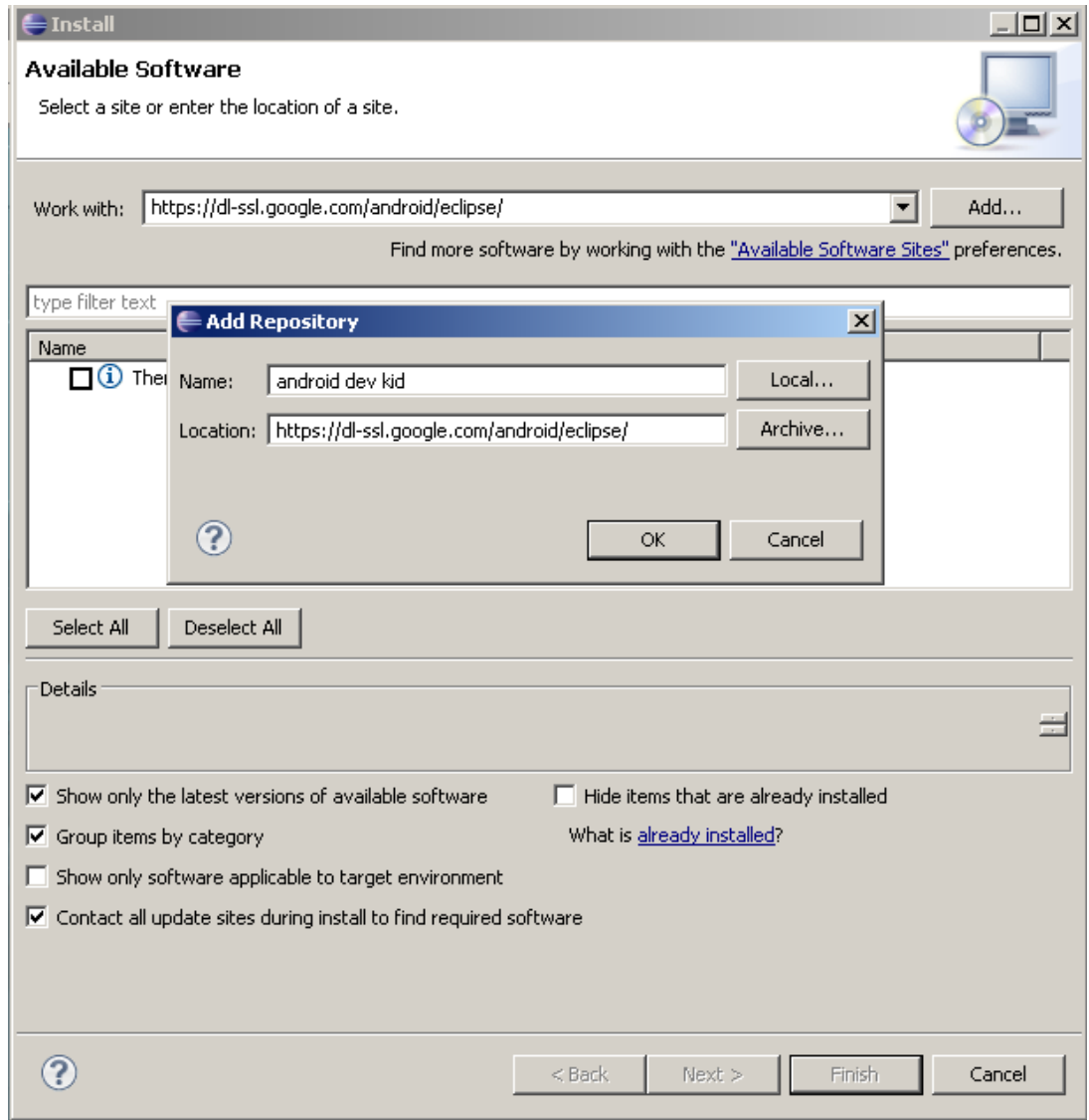


Install Packages denilir ve çıkan ekrandan Accept All diyerek Install edilir. Buradaki bekleme süresi internet hızı ile alakalıdır. Bağlantı yavaş ise uzun süre beklemek gerekebilir.



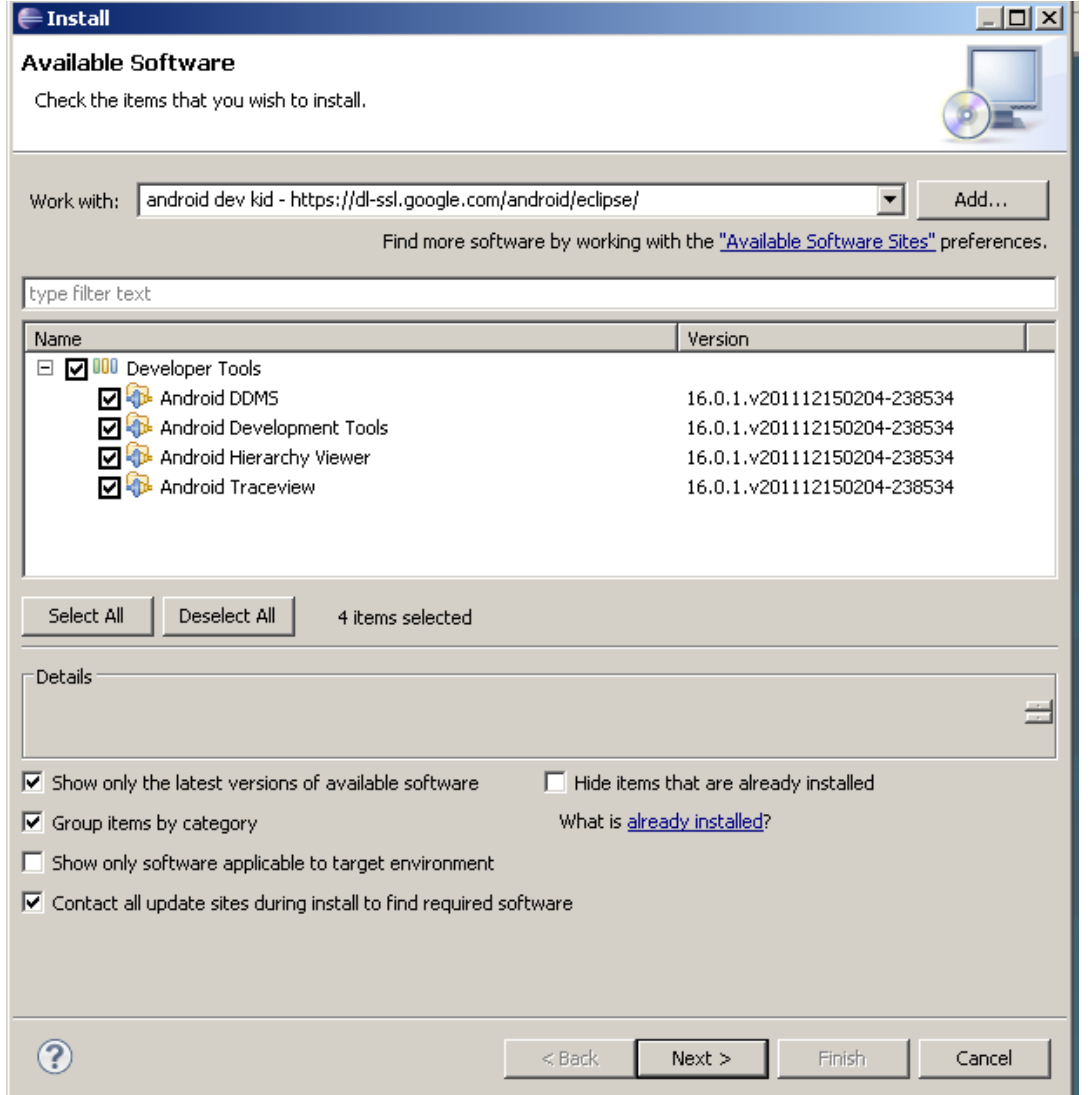
SDK kurulumundan sonra sonra, Eclipse ile Android projeyi açmasını sağlayacak olan Eclipsin ADT(Android Development Tools) eklentisini kurulacak. Bunun için Eclipse'den **Help > Install New Software** bölümüne tıklanır. Açılan pencereden aşağıdaki linki eklenir. Add tıklanır ve gelen ekranda name yazan yere bir isim vererek Ok tuşuna basılır.

<https://dl-ssl.google.com/android/eclipse/>



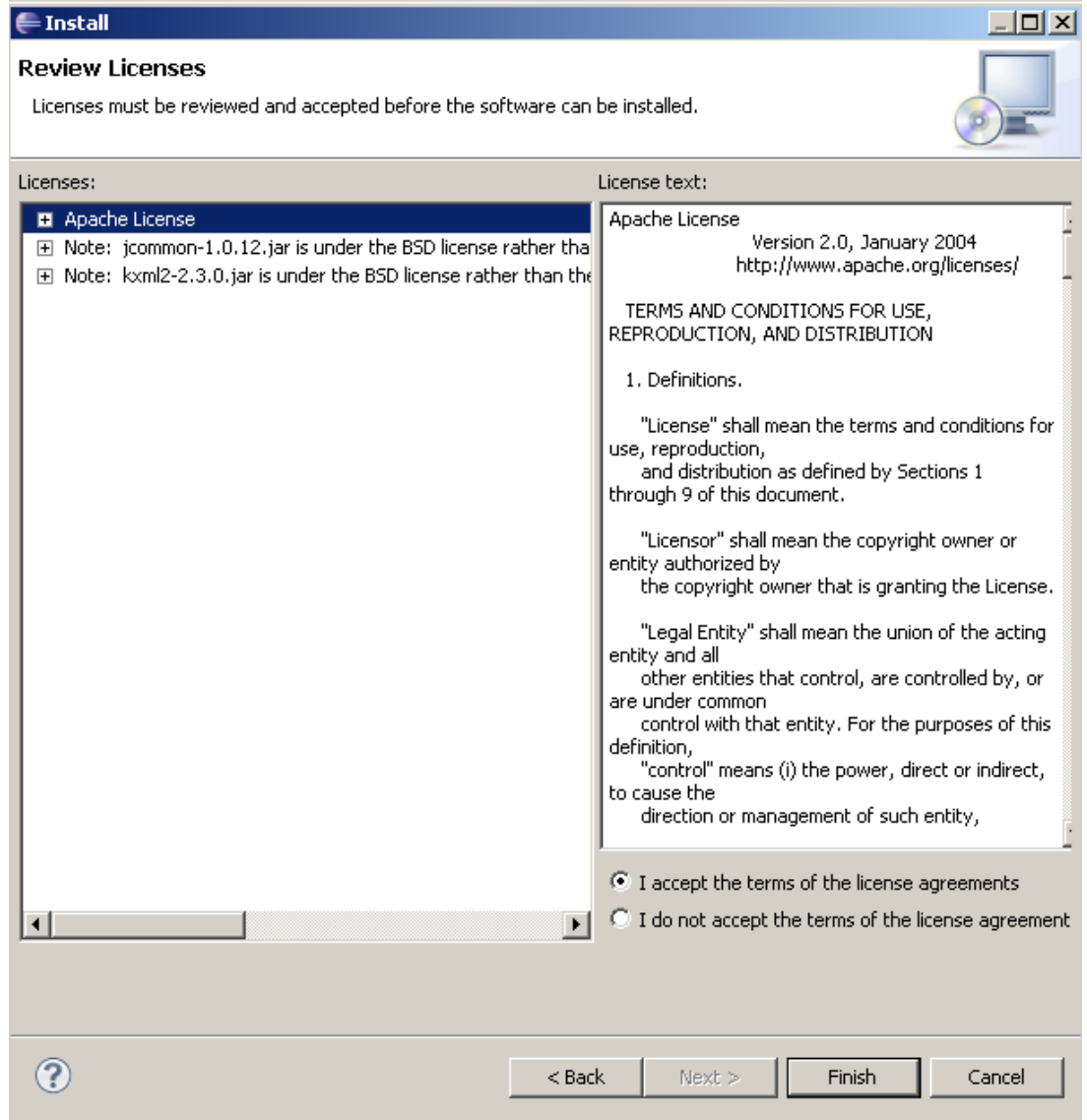


Ekledikten 5-10 sn sonra aşağıdaki resimdeki gibi Developer Tools ve altındakilerin gelmiş olması gerekir. Hepsini seçilerek NEXT denilir. Sonra ki gelen ekrandan bir daha NEXT denilir.



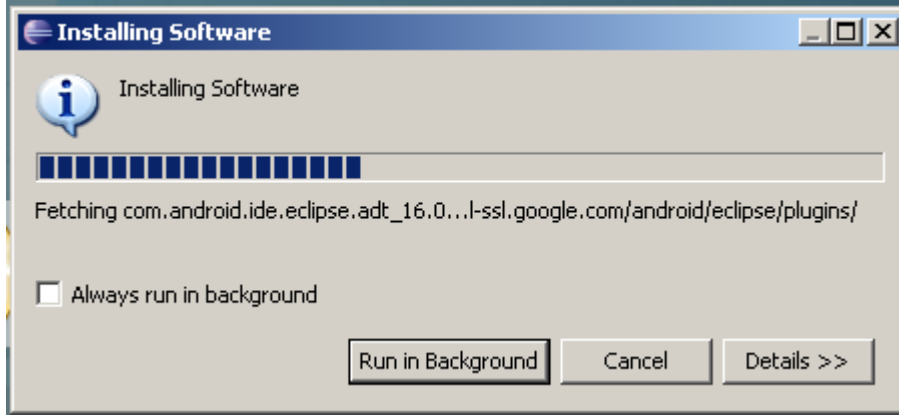


Aşağıdaki ekran geldiğinde lisans sözleşmelerini kabul ediliyor. FINİSH'e tıklanır.

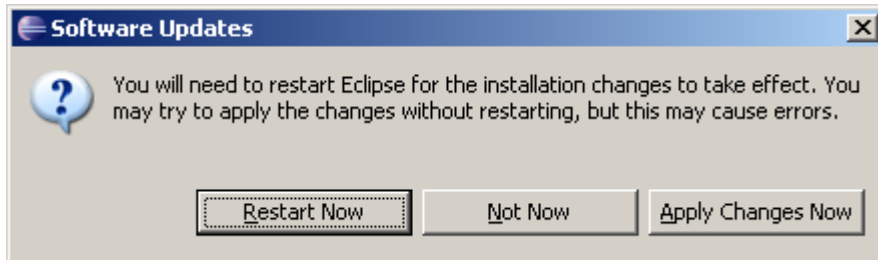




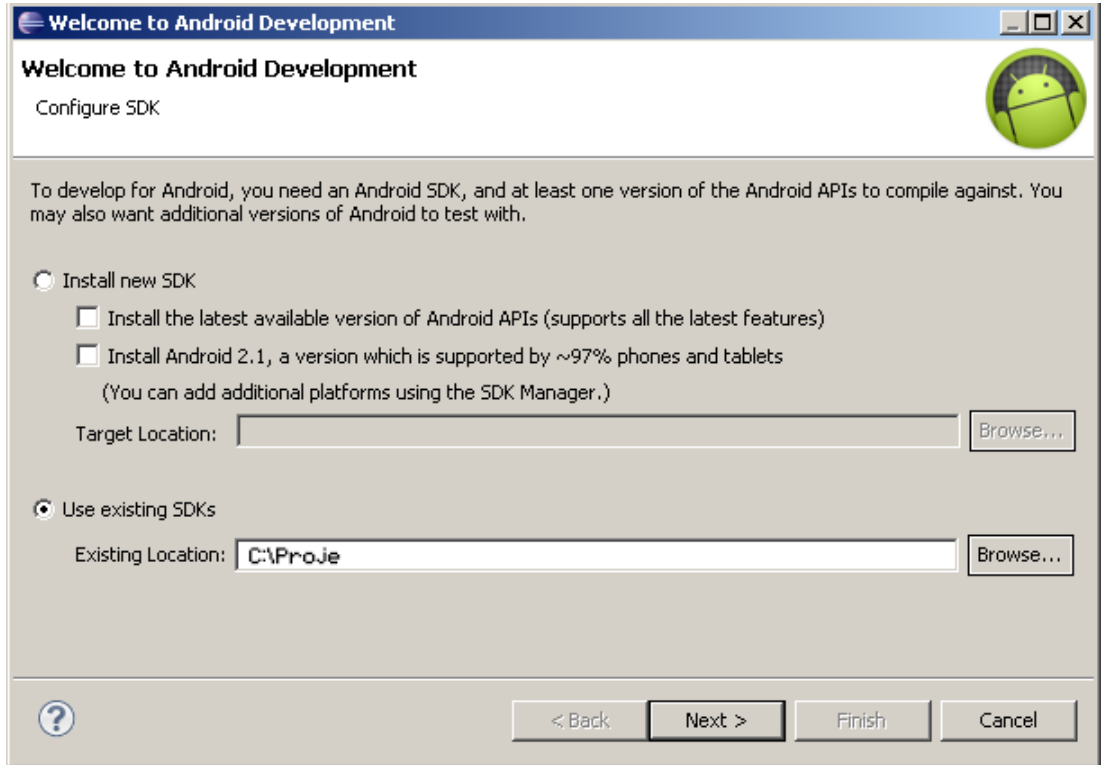
Buraya kadar bir problem yok ise ADT kurulumu başlamıştır.



Ve sonunda ADT'de kurulumu tamamlanır ve Restart Now diyerek Eclipse yeniden başlatılır.



Reset attığımızda karşımıza SDK ile ilgili ayarlar gelir. Oradan Use Existing SDKs seçeneğini seçilir. Android SDK'nın kurulu olduğu dizin seçilir.

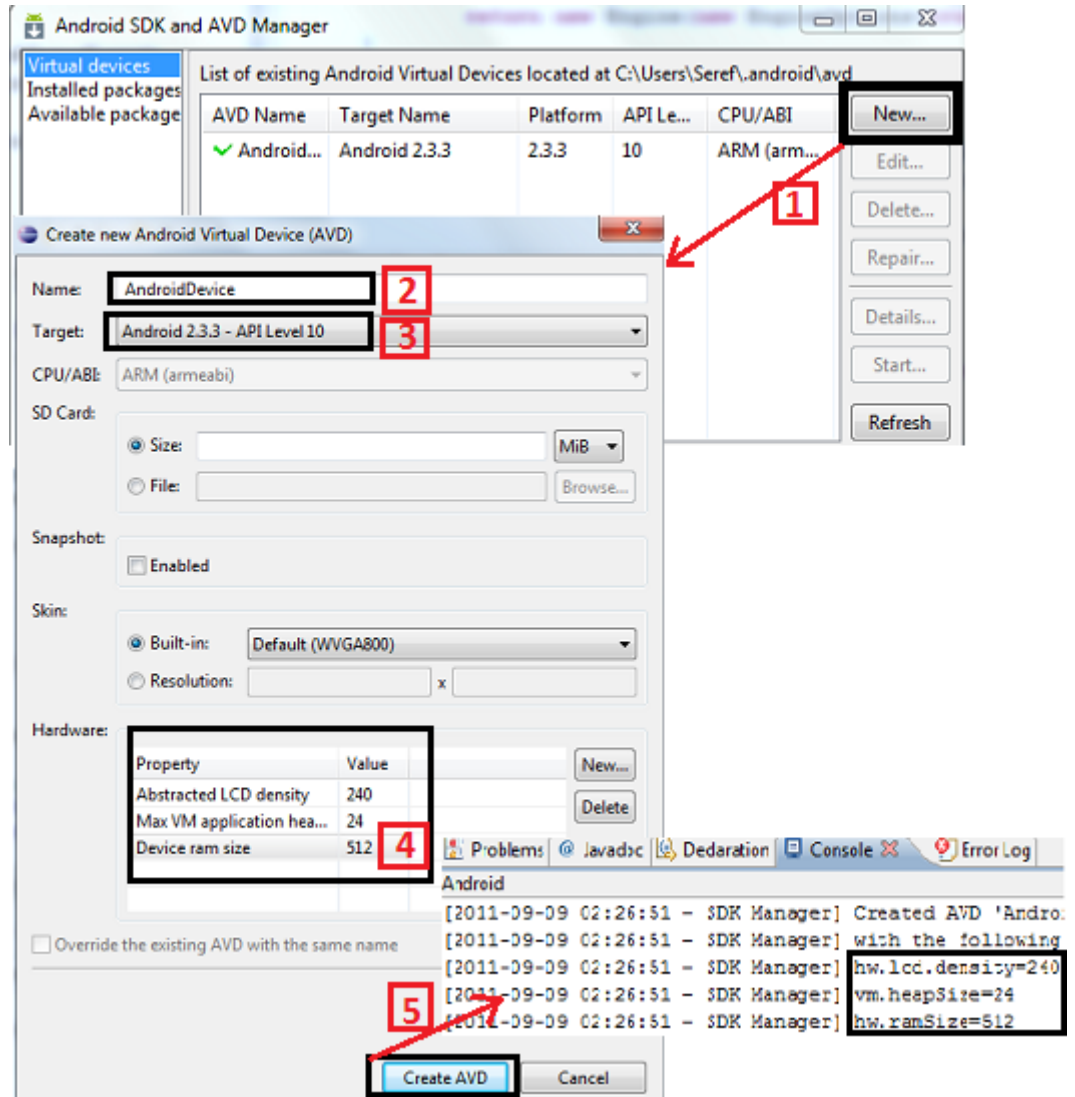


Ben yukarıdaki resimde seçmemişim fakat ALttaki seçeneği seçip Android SDK yi nereye kurduysanız o klasörü seçin. Örneğin benimki = **C:\Program Files\Android\android-sdk** ' da.

Not: SDK'nın bulunduğu dizini seçmekte sıkıntı yaşadıysanız, Eclipse'den Window-> Preferences -> Açılan ekrandan Android seçerek, SDK location yazan bölümden SDK'nın bulunduğu dizini güncelleyebilirsiniz.

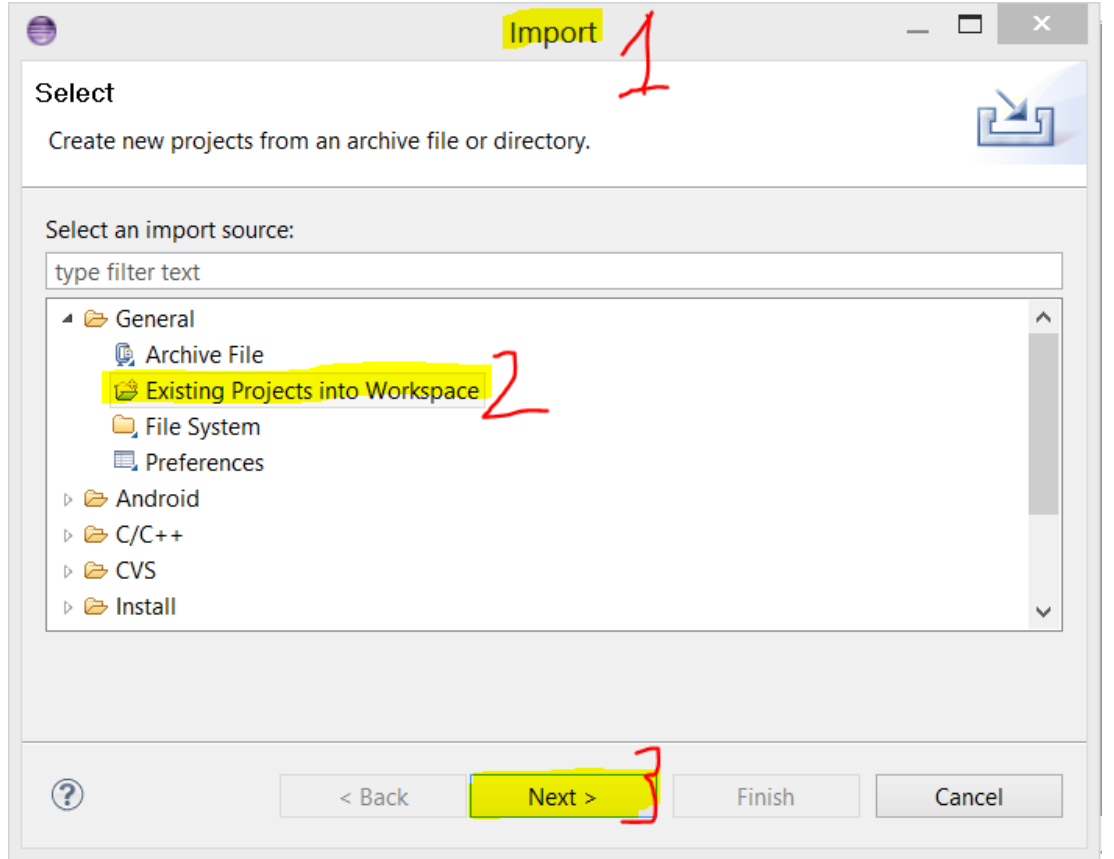


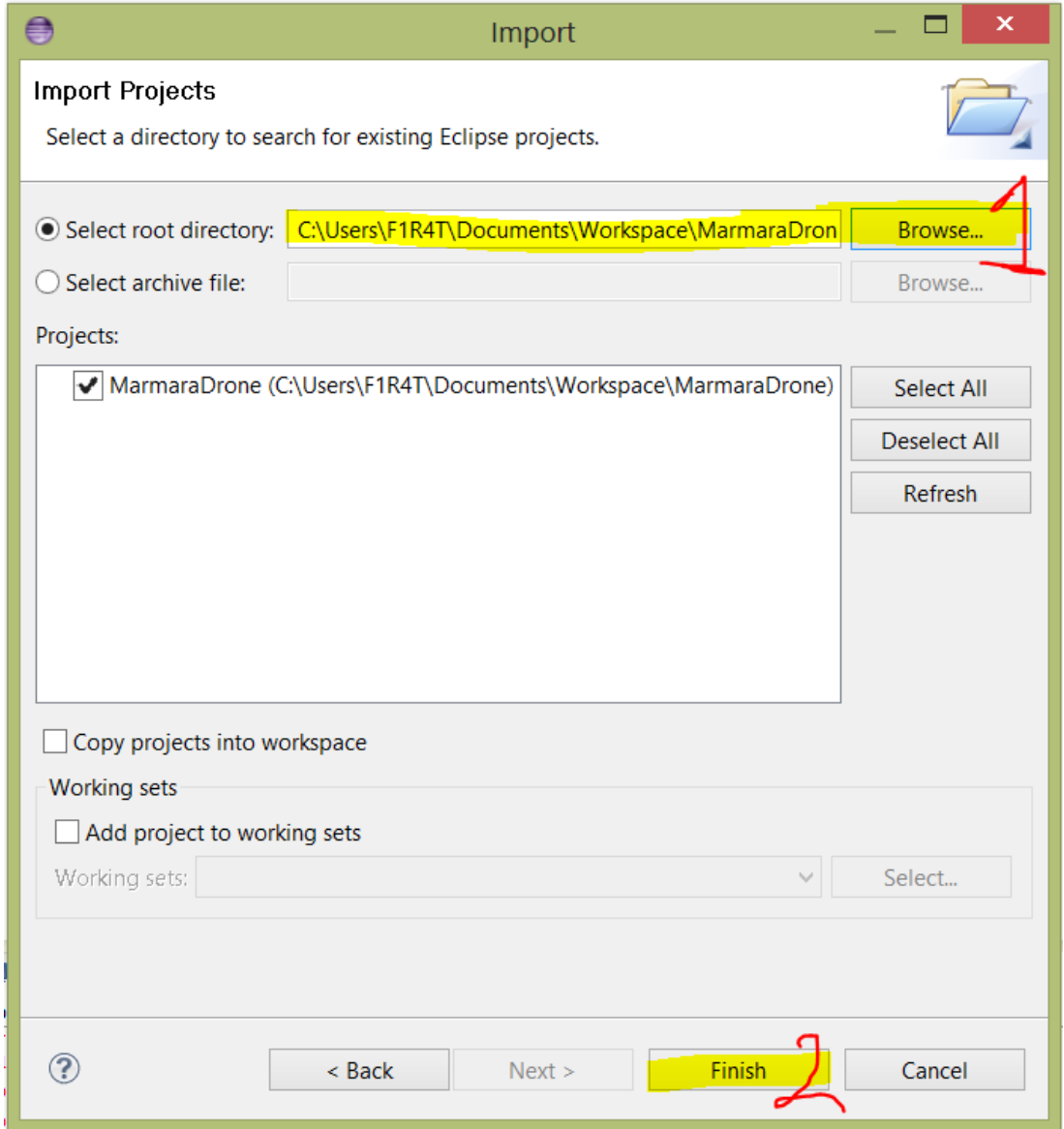
Uygulamayı çalıştırabilmek için sanal bir aygıtı ihtiyaç var. Bu sanal cihazı da yine eclipse içinde window → Android SDK and AVD Manager yolu ile açılan pencerenin sol bölümündeki **Virtual Devices** sekmesine tıklanıp. Sağ bölümde aşağıdaki pencereyi görülür. Bu pencereden ayarlamaları yapıldıktan sonra **Create AVD** tuşu ile sanal Cihaz oluşturma işlemi tamamlanır. Konsolda oluşturulan sanal aygıtın özelliklerini görebilir. Aşağıda bu paragrafın özeti niteliğinde resimlerden oluşan anlatımı görebilirsiniz.





Son olarak projeyi Eclipse eklemek için File menüsü altından İmport seçeneğiyle workspaceden dosya seçip sihirbazın tamamlamamız yeterli.





Resimlerdeki gibi projeyi en son Finish düğmesine tıklayarak proje ekleme işlemi tamamlanır.

Yararlanılan Kaynaklar

- ❖ AR.Drone Development SDK
<https://projects.ardrone.org/projects/show/ardrone-api>
- ❖ AR.Drone Parrot Support
<http://ardrone2.parrot.com/support/>
- ❖ AR.Drone Tanımı ve Özellikleri
<http://f1r4t.com/parrot-ar-drone-2-0-android-ios-uyumlu>
- ❖ Android Programlama Kitabı
Muharrem TAÇ
- ❖ CHIP Android ile Uygulama Geliştirme Eğitimi
Infinity Teknoloji
- ❖ Shigeo Yoshida Form Panelleri
<http://www.shigeodayo.com/>
- ❖ Codeminders Kütüphaneleri
<http://www.codeminders.com/>





AR.Drone Kamerasından





Fırat BİŞKİN

Sıcak bir yaz ayı olan Temmuz un 25 ‘ inde İstanbul / 4.Levent’ de Dünyaya Geldim. İsmet Aktar A.T.L Bilişim Teknolojileri Web Programlama branşı mezunu olup, Marmara Üniversitesi Teknik Bilimler Meslek Yüksekokulu Bilişim Teknolojileri Bölümünün Bilgisayar Programcılığı Programının son sınıf öğrencisiyim. Lise stajlarımın birini HP Gold Partneri Artı ve Artı Firmasında bir diğerini ise Çalık Holding de yaptım. Üniversite stajımı da Turkish Technic’de yapacağım.

Mehmet SABANCIOĞLU

1992 yılının ekim ayının 9’nda dünyaya gelmişim. Taşdelen İMKB A.T.L Bilişim Teknolojileri Veri Tabanı branşından mezun oldum.Lise Stajlarımı IBM Global Services ve IBM Turk’te yaptım. Marmara Üniversitesi Teknik Bilimler Meslek Yüksekokulu Bilişim Teknolojileri Bölümünün Bilgisayar Programcılığı Programının son sınıf öğrencisiyim.

Saygılar,

Fırat BİŞKİN

Mehmet SABANCIOĞLU