

## PENGEMBANGAN APLIKASI ANDROID MENGUNAKAN FIREMONKEY

Hendy Djaja Siswaja

Sekolah Tinggi Manajemen Informatika dan Komputer LIKMI  
Jl. Ir. H. Juanda 96 Bandung 40132

[hendy.djaya.s@gmail.com](mailto:hendy.djaya.s@gmail.com)

---

### ABSTRAK

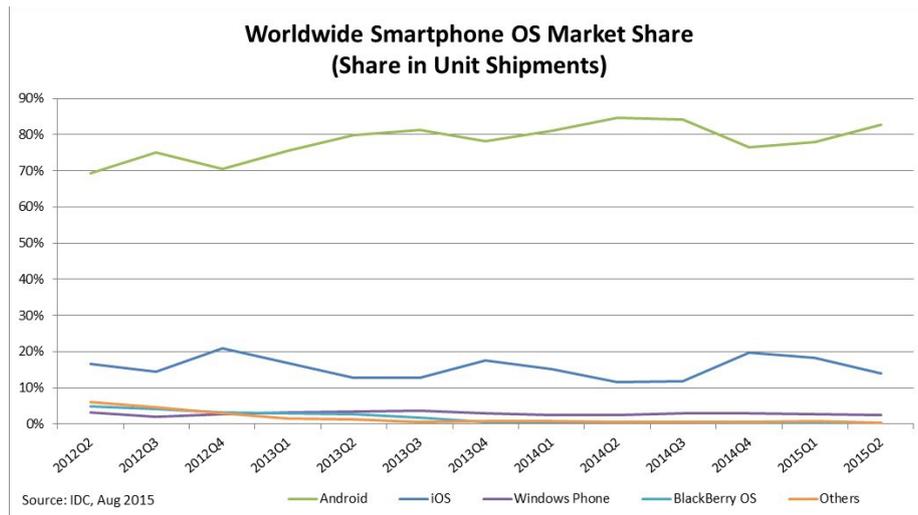
*Android merupakan sistem operasi smartphone yang bersifat open source. Android tidak terbatas pada satu merk tertentu namun dapat digunakan oleh banyak vendor sebagai sistem operasi smartphone mereka. Oleh karena itu, Android berhasil merajai pangsa pasar smartphone dunia berdasarkan sistem operasi yang digunakan pada smartphone tersebut. Melihat fakta ini, merupakan suatu peluang yang sangat baik untuk mengembangkan aplikasi mobile untuk sistem operasi Android. Namun, pada developer menghadapi tantangan dalam mengembangkan aplikasi mobile berbasis Android karena harus mempelajari bahasa pemrograman baru dan juga adaptasi dengan gaya pemrograman untuk Android ini. Embarcadero menjawab tantangan ini dengan meluncurkan framework FireMonkey sejak Embarcadero RAD Studio XE2 untuk mengembangkan aplikasi multi-platform, termasuk Android dan iOS dengan bahasa pemrograman pascal atau C++.*

**Kata-kata kunci** : android, pascal, multi-platform, framework, firemonkey.

---

### 1. PENDAHULUAN

Android merupakan sebuah sistem operasi open source untuk smartphone yang banyak digunakan oleh berbagai vendor, mulai dari yang ternama sampai vendor-vendor lokal. Android banyak diminati karena gratis dan dapat digunakan tanpa membayar royalti pada pemiliknya yaitu Google. Karena Android didukung penuh oleh Google dan dapat digunakan secara gratis oleh berbagai vendor, Android memiliki pangsa pasar sangat besar secara global. Menurut IDC (International Data Corporation), pada kuartal ke 2 tahun 2015, sistem operasi Android berhasil meraih 82,8% pangsa pasar dunia, disusul oleh iOS yang berhasil meraih 13,9% pangsa pasar dunia <http://www.idc.com/prodser/smartphone-os-market-share.jsp>).



Gambar 1.1 Pasang Pasar Sistem Operasi Smartphone Dunia

Berdasarkan data di atas, dapat dilihat bahwa peluang sangat terbuka lebar untuk mengembangkan aplikasi mobile berbasis Android. Tipe aplikasi mobile berbasis Android yang dapat dikembangkan dapat sangat bervariasi, mulai dari hiburan sampai dengan bisnis. Perusahaan-perusahaan besar pasti juga memikirkan hal yang sama dimana mereka membutuhkan developer-developer yang dapat mengembangkan aplikasi mobile berbasis Android untuk layanan yang mereka sediakan sehingga para pelanggan akan mendapatkan kemudahan akses. Selain itu, ketika sebuah aplikasi mobile berbasis Android dibuat dan masuk ke PlayStore (sarana untuk mendownload aplikasi-aplikasi mobile berbasis Android dari Google) maka target pasar dari pengguna aplikasi tersebut akan bertambah besar menjadi skala internasional karena melalui PlayStore ini, seluruh pengguna sistem operasi Android di seluruh dunia dapat mendownload aplikasi tersebut.

Namun, mengembangkan aplikasi mobile berbasis Android ini bukan tanpa tantangan. Tantangan dan hambatan utamanya adalah programmer yang sebelumnya tidak pernah menggunakan bahasa pemrograman Java harus mempelajari dan beradaptasi dengan bahasa pemrograman ini karena pada dasarnya Android menggunakan bahasa pemrograman Java. Bagi programmer, bahasa pemrograman layaknya bahasa keseharian, analoginya orang yang sehari-hari menggunakan bahasa Indonesia untuk berkomunikasi pasti membutuhkan waktu untuk mempelajari bahasa Inggris untuk berkomunikasi dan juga meskipun sudah mempelajari bahasa Inggris, butuh waktu yang lebih panjang lagi untuk benar-benar menguasai dan terbiasa dengan bahasa Inggris.

Embarcadero sebagai perusahaan yang memiliki IDE (Integrated Development Environment) Delphi yang menggunakan bahasa pemrograman Pascal memberikan

jawaban terhadap hambatan tersebut di atas dengan meluncurkan framework FireMonkey yang sangat membantu para programmer yang terbiasa dengan bahasa pemrograman Pascal untuk mengembangkan aplikasi mobile berbasis Android. Dengan adanya framework FireMonkey ini, programmer tidak perlu mempelajari bahasa pemrograman baru tapi tetap perlu sedikit beradaptasi dengan beberapa fitur baru untuk pengembangan aplikasi mobile berbasis Android.

## **2. DEFINISI**

### **2.1. Android**

*“Android is a comprehensive open source platform designed for mobile devices. It is championed by Google and owned by Open Handset Alliance. The goal of the alliance is to “accelerate innovation in mobile and offer consumers a richer, less expensive, and better mobile experience.””* (Gargenta, 2011:1) yang dapat diartikan sebagai berikut :

*“Android adalah platform open source yang komperhensif yang didesain untuk perangkat mobile. Android diprakasai oleh Google dan dimiliki oleh Open Handset Alliance (dengan Google sebagai salah satu anggotanya). Android dibuat dengan tujuan untuk mempercepat inovasi perangkat mobile dan menawarkan kepada pelanggan pengalaman penggunaan perangkat mobile yang lebih kaya, lebih murah dan lebih baik.”*

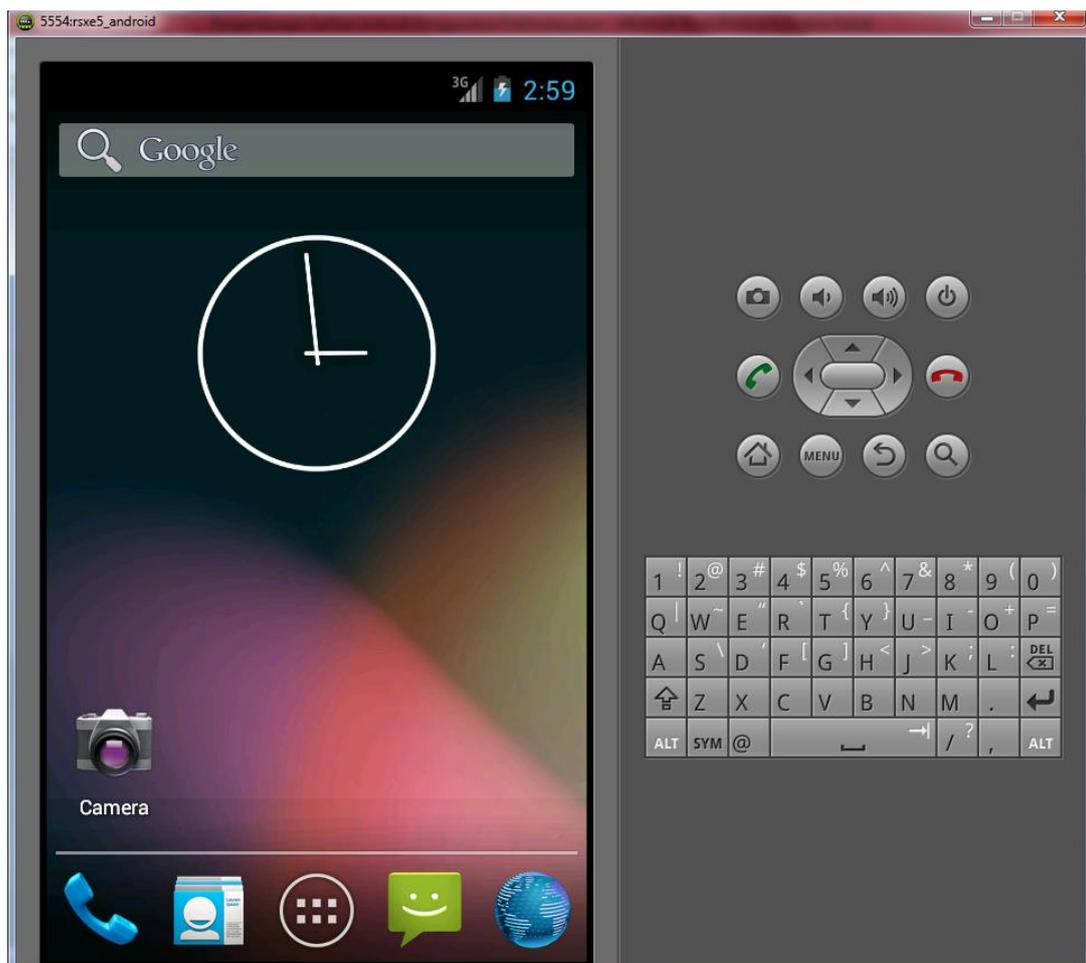


Gambar 2.1 Logo Android

Android merupakan sebuah revolusi dibidang perangkat mobile. Android merupakan sistem terbuka pertama yang memisahkan perangkat lunak dari perangkat kerasnya. Hal ini sudah biasa ditemukan pada komputer PC dimana berbagai merk PC dapat dipasang berbagai sistem operasi tapi pada perangkat mobile, hal ini adalah yang pertama. Dengan adanya revolusi ini, Android dapat dipasang pada berbagai perangkat mobile tanpa bergantung pada merk tertentu. Android juga dapat terus dikembangkan,

diperbaharui dan ditingkatkan sesuai dengan kesiapan perangkat keras yang dapat disediakan oleh para vendor.

Android adalah platform yang komperhensif, artinya Android dibuat sebagai satu paket lengkap perangkat lunak yang digunakan pada perangkat lunak. Android dibuat satu paket lengkap sehingga sangat memudahkan pengembang, pengguna, dan vendor. Bagi pengembang, Android menyediakan semua tools dan framework yang dibutuhkan untuk mengembangkan aplikasi mobile berbasis Android. Pengembang hanya membutuhkan Android SDK (Software Development Kit) untuk mulai mengembangkan aplikasi berbasis Android ini bahkan pengembang tidak membutuhkan perangkat keras smartphome untuk menguji coba aplikasi yang dikembangkan karena Android SDK ini juga sudah menyediakan emulator atau simulasi smartphome yang dapat berjalan hampir sama dengan smartphome sesungguhnya. Keterbatasan dari emulator ini adalah dari pembacaan hardware sensor seperti gyroscope, compass, temperature, dan lai-lain yang hanya ada pada perangkat keras smartphome sesungguhnya.



Gambar 2.2 Tampilan Emulator Android

Android sendiri memiliki sejarah yang cukup menarik dalam masa pengembangannya. Marko Gargenta dalam bukunya *Learning Android*, menuliskan sejarah Google sebagai berikut :

1. *In 2005, Google buys Android, Inc. The world thinks a “gPhone” is about to come out.*
2. *Everything goes quiet for a while.*
3. *In 2007, the Open Handset Alliance is announced. Android is officially open sourced.*
4. *In 2008, the Android SDK 1.0 is released. The G1 phone, manufactured by HTC and sold by the wireless carrier T-Mobile USA, follows shortly afterward.*
5. *2009 sees a proliferation of Android-based devices. New versions of the operating system are released: Cupcake (1.5), Donut (1.6), and Eclair (2.0 and 2.1). More than 20 devices run Android.*
6. *In 2010, Android is second only to Blackberry as the best-selling smart phone platform. Froyo (Android 2.2) is released and so are more than 60 devices that run it. (Gargenta, 2011:3)*

Ketika Google membeli Android Inc. di tahun 2005, dunia berspekulasi bahwa Google akan meluncurkan smartphone produksi sendiri yang diberi nama “gPhone”, namun spekulasi ini terbantahkan pada tahun 2007 ketika Open Handset Alliance dibentuk dan Android diumumkan secara resmi sebagai perangkat lunak open source. Tujuan utama dari sistem operasi Android, bukanlah untuk satu jenis perangkat mobile saja tapi lebih besar dari itu, Android ditujukan sebagai platform open source yang dapat digunakan diberbagai perangkat dan boleh diubah, disesuaikan dan ditingkatkan oleh vendor manapun agar dapat berjalan dengan baik pada perangkat keras yang mereka miliki. Saat ini, Android sudah berada di versi 6.0, Code Name : Marshmallow (<https://www.android.com/history/#/marshmallow>)

## **2.2.FireMonkey**

*“The FireMonkey FMX framework is the app development and runtime platform behind Delphi, C++ Builder, and Appmethod. FireMonkey is the first native GPU-powered application platform.” (Teti, 2014:91)*

Yang dapat diartikan sebagai berikut :

*“Framework FireMonkey FMX adalah platform pengembangan dan runtime aplikasi pada Delphi, C++ Builder, dan Appmethod. FireMonkey adalah platform aplikasi pertama yang asli berbasis GPU (Graphic Processing Unit).”*

Dunia IT, hari demi hari semakin berkembang ke arah multiplatform. Pengembangan aplikasi berdasarkan satu jenis platform saja dianggap sudah tidak sesuai lagi pada masa kini. Hambatan utamanya adalah, karena berbeda platform, pada saat terjadi perubahan pada aplikasi maka pengembang harus melakukan perubahan diseluruh platform. Hal seperti ini akan sangat merepotkan dan meningkatkan peluang terjadinya kesalahan manusia (human error) seperti lupa dan lain sebagainya. FireMonkey merupakan terobosan baru dan teknologi kunci dari Embarcadero untuk menjawab kebutuhan pengembangan aplikasi multiplatform dan multidevice. FireMonkey didesain untuk pengembangan aplikasi native untuk platform MS Windows, Mac OS X, Android, dan iOS.

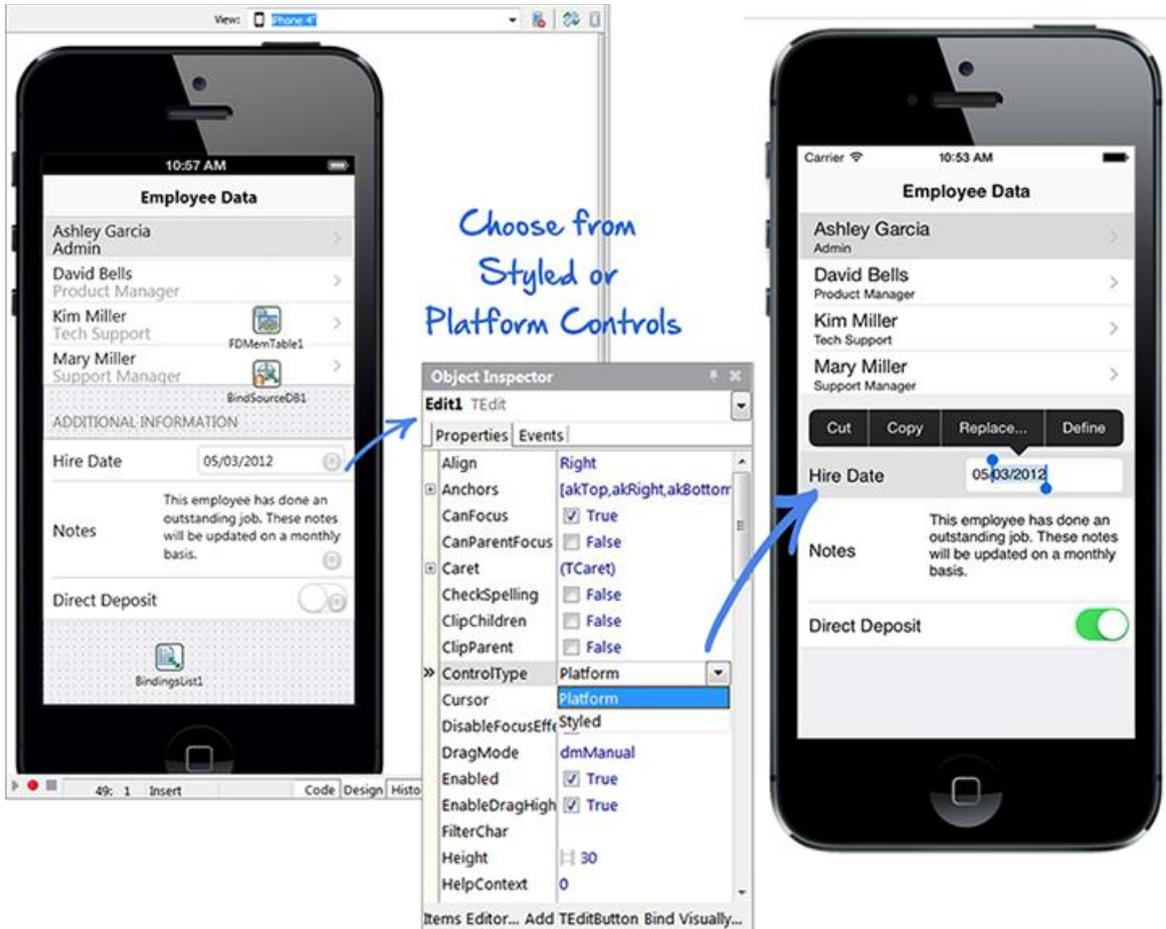
Framework FireMonkey ini diperkenalkan oleh Embarcadero sejak RAD Studio XE2 dan para pengembang yang terbiasa dengan IDE (Integrated Development Environment) Delphi akan sangat terbantu dengan adanya framework FireMonkey ini karena pengembang menggunakan bahasa pemrograman yang sama yaitu bahasa Pascal untuk mengembangkan aplikasi berbagai platform yang telah disebutkan di atas. Pengembang dapat membuat satu aplikasi dan mengkompilasi aplikasi tersebut utk berbagai platform yang diinginkan. Ketika terjadi perubahan, hanya satu aplikasi saja yang perlu diubah, dengan demikian, kesalahan dapat lebih dikurangi.



Gambar 2.3 Logo FireMonkey

Salah satu fitur unggulan dari FireMonkey yang memungkinkan FireMonkey dapat membangun aplikasi multiplatform adalah styled component. Component-component dapat dikostumasi berdasarkan style yang diinginkan atau sesuai dengan platform yang

ingin digunakan. Dengan melakukan perubahan style ini, aspek dari component dapat diubah namun kode program tidak perlu diubah. Relasi seperti ini sama seperti relasi antara HTML dan CSS, dimana yang satu adalah konten sementara yang lainnya adalah tampilan.



Gambar 2.4 Styled Components

Embarcadero menyatakan juga bahwa menggunakan FireMonkey ini, pengembang benar-benar mengembangkan aplikasi native untuk masing-masing platform, bukan aplikasi yang seolah-olah native padahal menggunakan script tambahan atau emulator atau virtual machine untuk dapat berjalan. Yang menjadi ciri dari aplikasi native yang sebenarnya menurut Embarcadero adalah sebagai berikut :

1. Fast.

Aplikasi native dapat berjalan cepat karena langsung berjalan di CPU perangkat keras dan tidak dibatasi atau diperlambat oleh penerjemahan script ataupun mesin virtual.

2. Predictable.

Aplikasi native berjalan langsung di CPU perangkat keras sesuai dengan spesifikasi dari vendor dan tidak diperlambat oleh software layers tambahan atau proses tambahan.

3. Better User Experience (UX).

Aplikasi native berjalan dengan sepenuhnya memanfaatkan kemampuan dan performa dari perangkat mobile.

### **3. PENGEMBANGAN APLIKASI ANDROID DENGAN MENGGUNAKAN FIREMONKEY**

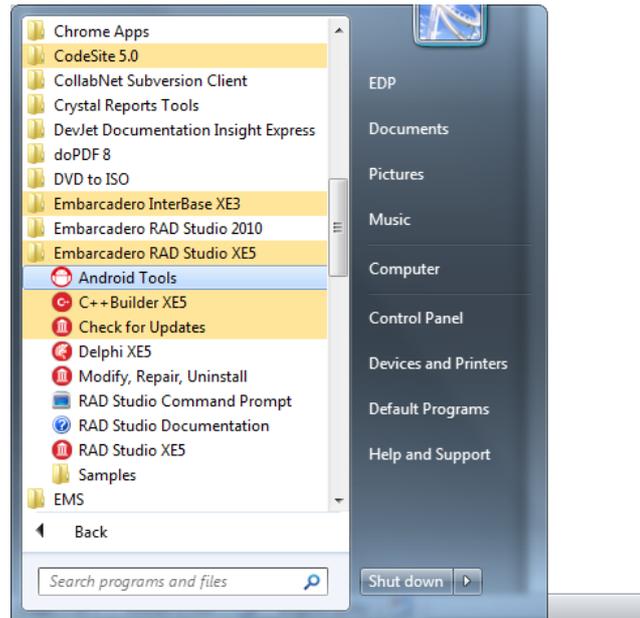
Embarcadero sejak RAD Studio XE2 telah menyediakan framework FireMonkey sebagai bagian dari Delphi atau C++ Builder. Embarcadero memisahkan framework FireMonkey ini berdasarkan tipe aplikasi yang akan dikembangkan, aplikasi Desktop atau aplikasi Mobile. Aplikasi Desktop dapat digunakan untuk membangun aplikasi Desktop untuk platform MS Windows dan Mac OS X, sedangkan aplikasi Mobile dapat digunakan untuk membangun aplikasi untuk platform Android dan iOS.

Untuk platform Android, Embarcadero telah menyediakan Android SDK dan juga Android NDK (Native Development Kit) untuk bahasa pemrograman C/C++. Pada jurnal ini, akan digunakan RAD Studio XE5 dengan Android SDK versi 4.2.2 (Code Name : Jelly Bean).

#### **3.1. Menjalankan AVD (Android Virtual Device) atau Emulator Android**

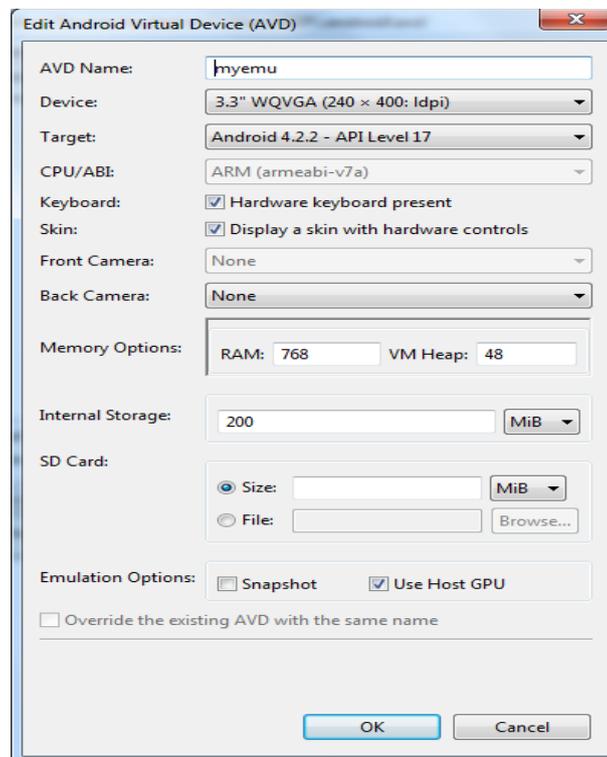
Pada saat mengembangkan aplikasi Android, AVD atau emulator Android harus dijalankan terlebih dahulu karena pada saat compiling, aplikasi mobile akan langsung diinstall pada emulator yang telah dipilih. Sebenarnya, tanpa menjalankan emulator terlebih dahulu pun Delphi akan meminta Anda untuk menjalankan AVD pada saat proses compiling namun mengingat proses booting dari emulator ini memakan waktu yang cukup lama, maka ada baiknya jika dijalankan terlebih dahulu.

Untuk menjalankan AVD, silakan buka Android Tools yang sudah terinstall bersamaan dengan ketika RAD Studio XE5 diinstall.



Gambar 3.1 Menu Android Tools pada Start Windows

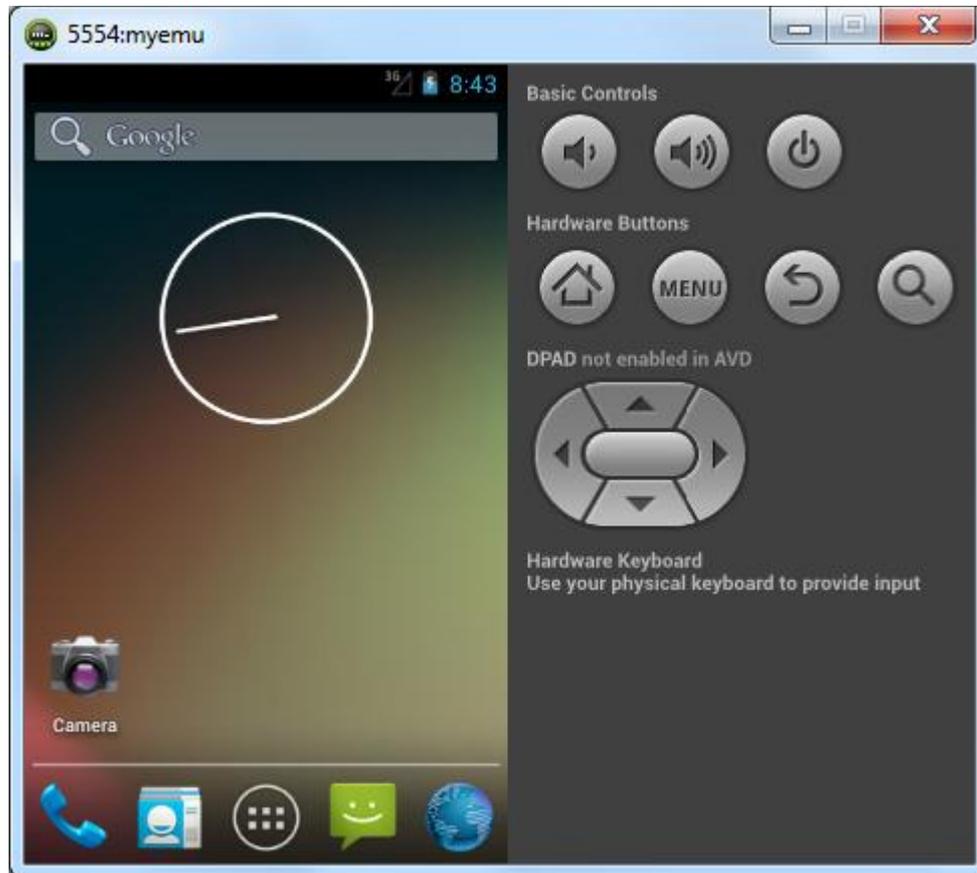
Setelah terbuka, pilih menu Tools > Manage AVDs. Pada jendela ini, akan muncul daftar AVD yang dimiliki dan dapat digunakan. Secara default, RAD Studio XE5 sudah menyiapkan sebuah AVD yakni, rsxe5\_android, namun sangat disarankan sekali untuk membuat AVD baru yang lebih sesuai dengan spesifikasi hardware komputer. Sebagai contoh adalah sebagai berikut :



Gambar 3.2 Contoh konfigurasi AVD baru

AVD yang baru menggunakan tipe device 3,3” WQVGA agar loading lebih cepat dan tidak membebani kinerja komputer terlalu berat.

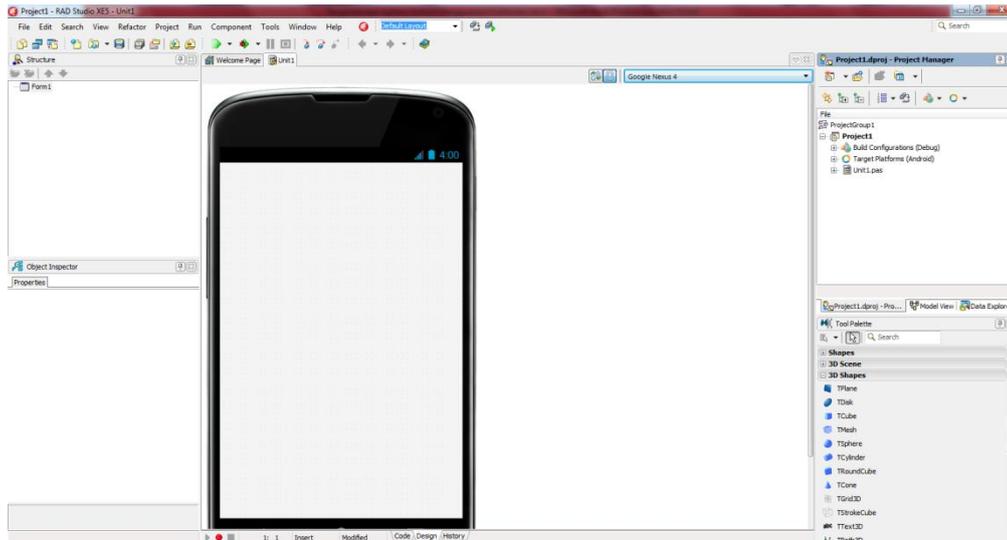
Setelah AVD baru berhasil dibuat, silakan klik pada AVD yang sudah dibuat dan tekan tombol Start, kemudian Launch untuk menjalankan AVD. Tunggu sampai jendela AVD muncul dan sistem operasi Android berjalan.



Gambar 3.3 Tampilan AVD 3,3” WQVGA

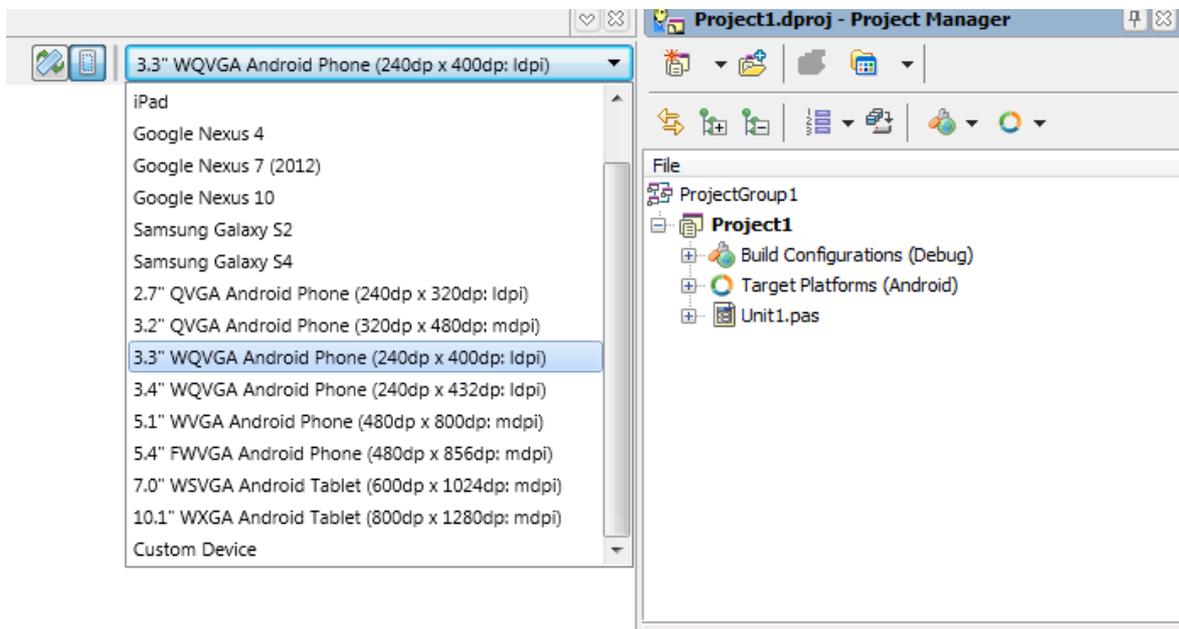
### 3.2. Membuat Aplikasi Android dengan FireMonkey

Setelah AVD berjalan, masuk ke Delphi XE5 dan pilih menu File > New > FireMonkey Mobile Application – Delphi. Project baru akan terbuka dan tampilan form akan muncul seperti sebuah smartphone Android.



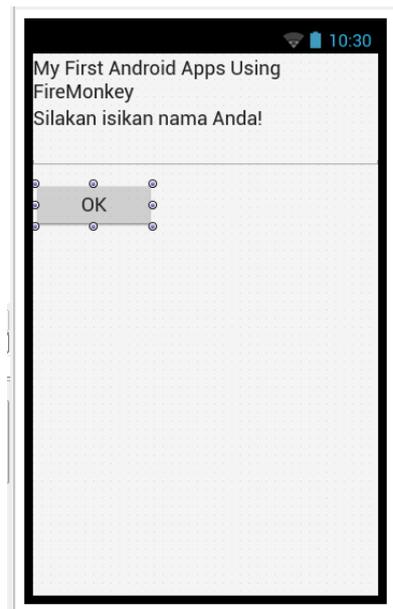
Gambar 3.4 Tampilan Project FireMonkey Mobile Application

Langkah berikutnya adalah menyesuaikan template dengan AVD yang sudah dibuat sebelumnya. Karena pada contoh di atas AVD yang dibuat menggunakan 3,3” WQVGA, maka project harus disesuaikan dengan AVD tersebut, jika hal ini tidak dilakukan maka mungkin saja tampilan aplikasi akan terlalu besar atau terlalu kecil karena resolusi yang berbeda.



Gambar 3.5 Sesuaikan Template dengan AVD yang digunakan

Setelah itu, mulailah membuat aplikasi Android dengan menambahkan komponen-komponen yang dibutuhkan.



Gambar 3.6 Antar Muka Aplikasi

Selanjutnya isikan program yang diinginkan pada tombol OK dengan cara mengklik 2 kali tombol OK tersebut. Pada halaman kode, pemrograman dilakukan dengan bahasa Pascal. Bagi para pengembang yang sudah sangat terbiasa dengan Delphi dan bahasa Pascal, akan dengan sangat mudah beradaptasi.

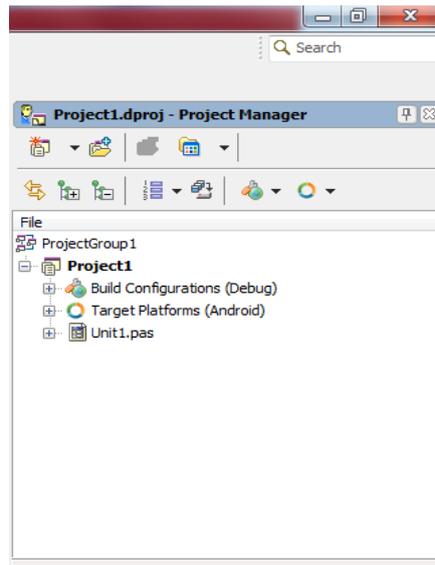
```

30 procedure TForm1.Button1Click(Sender: TObject);
   begin
       if Edit1.Text='' then
           ShowMessage('Nama tidak boleh kosong!!!')
       else
           ShowMessage('Selamat Datang, '+Edit1.Text+#13#13+
36 'FireMonkey menyambut Anda dengan gembira.');
```

Gambar 3.7 Contoh kode program untuk tombol OK

### 3.3. Menjalankan Aplikasi Android Pada AVD atau Emulator Android

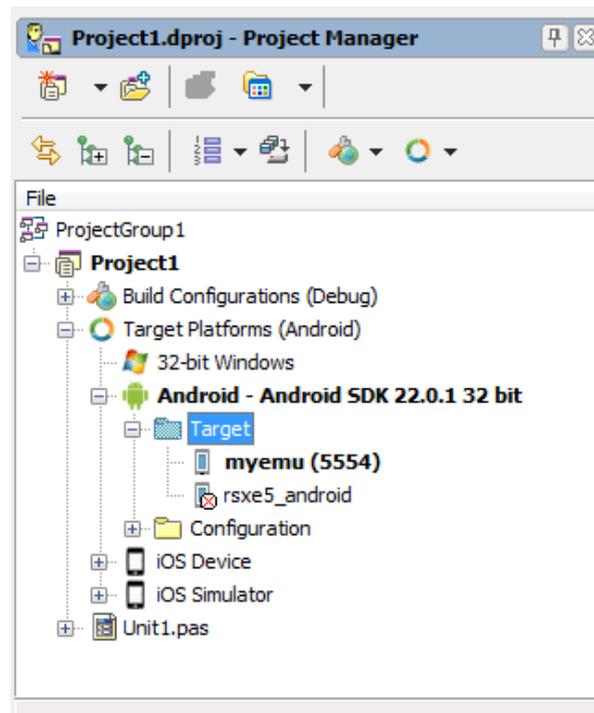
Setelah aplikasi Android telah dibuat sesuai dengan kebutuhan, langkah berikutnya ada menguji aplikasi tersebut pada AVD yang telah disiapkan sebelumnya. Untuk memilih AVD yang akan menjalankan aplikasi, lihat bagian Project Manajer, pada bagian ini, dapat ditemukan Target Platforms.



Gambar 3.8 Project Manager dan Target Platforms

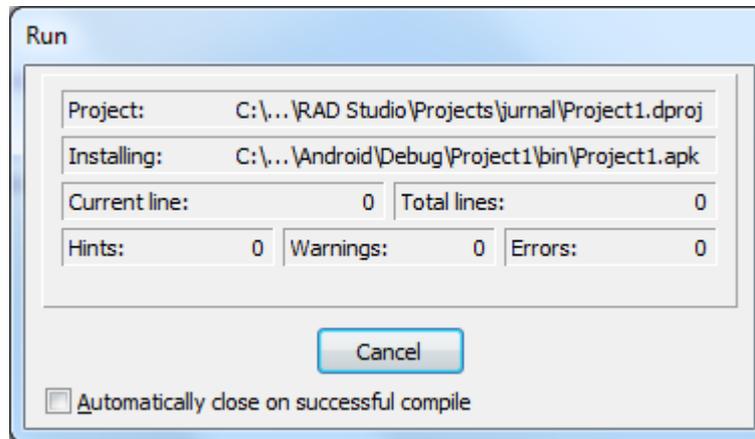
Dropdown Target Platforms untuk melihat platform yang tersedia, kemudian dropdown kembali Android, kemudian dropdown kembali Target. Seharusnya AVD yang sudah dijalankan sebelumnya akan muncul pada Target, jika belum ada, coba klik kanan pada Target dan pilih Refresh.

Target AVD atau emulator yang akan digunakan harus yang dicetak tebal, jika AVD yang telah disiapkan sebelumnya ada didaftar namun belum di cetak tebal, klik kanan pada AVD kemudian pilih Activate. Pastikan langkah ini telah dikerjakan sebelum mengompile dan menjalankan aplikasi.



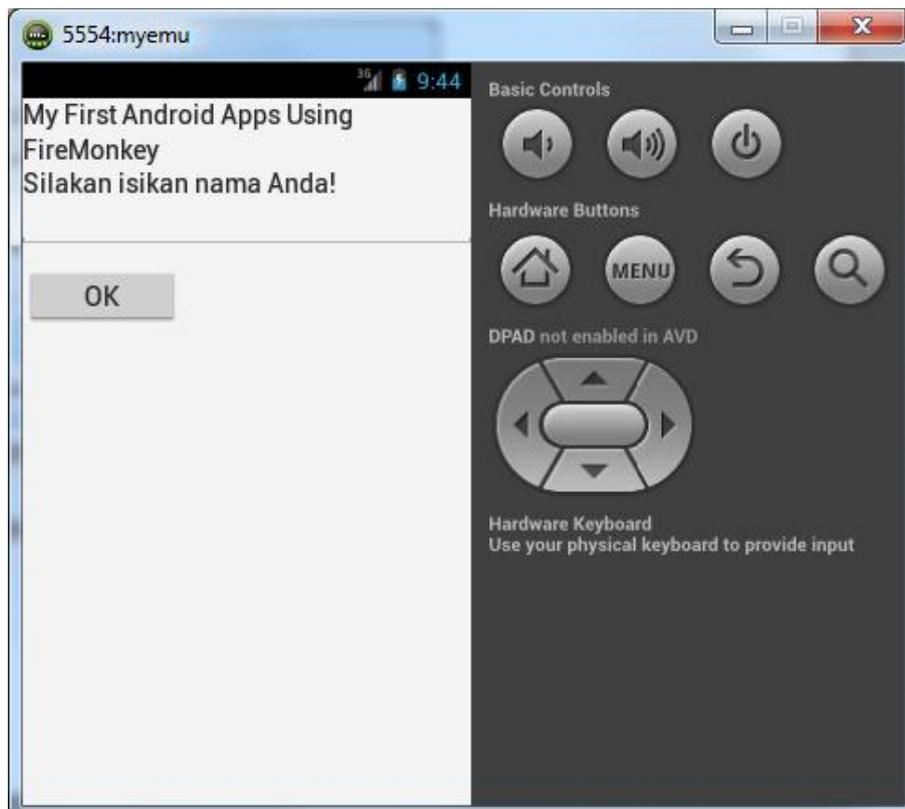
Gambar 3.9 Target AVD yang sesuai sudah terpilih

Setelah target AVD ditentukan, compile dan jalankan aplikasi dengan menggunakan Run Without Debugging (Shift+Ctrl+F9). Tunggu sampai proses compiling selesai dan aplikasi diinstall pada AVD yang telah disiapkan.



Gambar 3.10 Aplikasi sedang diinstall ke AVD

Buka kembali AVD yang telah disiapkan sebelumnya karena aplikasi setelah terinstall akan langsung dijalankan di AVD.



Gambar 3.11 Aplikasi Android sudah berjalan di AVD

Masukkan nama kemudian tekan tombol OK untuk menguji apakah pesan muncul atau tidak.



Gambar 3.12 Tampilan Pesan pada aplikasi

Untuk kembali ke halaman utama AVD, tekan tombol Home atau Back sampai kembali ke halaman utama.

FireMonkey sangat mudah digunakan bagi para pengembang yang sebelumnya menggunakan Delphi dan bahasa pemrograman Pascal. Walaupun ada beberapa hal yang perlu dipelajari lagi namun semua itu hanya membutuhkan waktu adaptasi yang cukup singkat.

#### 4. DAFTAR PUSTAKA

- [1]. Gargenta, Marko, *Learning Android*, O'Reilly, 2011.
- [2]. Teti, Daniele, *Delphi Cookbook*, Packt Publishing, 2014.
- [3]. Embarcadero, *Product Documentation RAD Studio Mobile Tutorials*, Embarcadero, 2014.
- [4]. <https://www.android.com/>
- [5]. <http://www.embarcadero.com/products/rad-studio/multi-device-true-native>