

# PENGEMBANGAN APLIKASI GUI MULTIGUNA BERBASIS ANDROID UNTUK DISPLAI DATA JARINGAN LOKAL DAN INTERNET

Pratikto <sup>1)</sup>, Raydha Zul Fitriani <sup>2)</sup>

<sup>1),2)</sup> Teknik Refrigerasi dan Tata Udara, Politeknik Negeri Bandung  
Jl. Gegerkalong Hilir, Ds. Ciwaruga Kotak Pos 1234 Bandung 40012  
Email : pratikto@hotmail.co.jp

**Abstrak .** *Android merupakan sistem operasi yang paling populer untuk smartphone yang kini digunakan untuk berbagai keperluan. Kunci pemanfaatan smartphone berada pada ketersediaan aplikasi di dalamnya. Berbagai aplikasi untuk beragam keperluan telah banyak dikembangkan baik yang dapat diunduh secara gratis maupun berbayar. Namun tidak jarang pengguna mengalami kesulitan untuk memanfaatkan smartphone sesuai dengan keperluan khusus yang mungkin bervariasi, misalnya mendisplaikan dan mengolah data, User Interface untuk peralatan tertentu dan mengkoneksikan perangkat dengan internet sehingga dapat diakses secara online. Di pihak lain terdapat juga server data gratis yang dapat dimanfaatkan untuk menyambungkan suatu perangkat ke internet. Dalam makalah ini dibahas pengembangan aplikasi multiguna pada smartphone yang dapat difungsikan dalam membangun GUI (Graphical User Interface) untuk berbagai keperluan akses data dari server data lokal dan internet, pengolahan dan penampilan data dalam berbagai bentuk. Aplikasi dirancang dapat diisi code HTML/Javascript dengan teknik Web Programming untuk keperluan khusus pengguna menggunakan tool editor yang disediakan. Fungsi dari aplikasi dapat dirancang dan dipilih sendiri oleh pengguna sebagaimana code dapat dibuat juga dengan perangkat Web editor lain. Fungsi dari aplikasi untuk mendisplaikan data dari server data SparkFun dapat ditampilkan dengan baik dan leluasa menggunakan library berbasis jQuery.*

**Kata kunci:** *smartphone, aplikasi multiguna, Web Programming.*

## 1. Pendahuluan

Maraknya penggunaan *smartphone* Android dijamin sekarang menyebabkan menjamurnya aplikasi berbasis Android. Android sendiri merupakan sistem operasi yang paling populer untuk *smartphone* yang digunakan untuk berbagai keperluan. Hal tersebut sehingga menjamurnya aplikasi yang menunjang berbagai keperluan pengguna. Namun tidak jarang pengguna mengalami kesulitan dalam menggunakan aplikasi yang ditujukan untuk keperluan khusus dalam *smartphone* tersebut, misalnya mendisplaikan dan mengolah data, *User Interface* untuk peralatan tertentu dan mengkoneksikan perangkat dengan internet sehingga dapat diakses secara online.

Server data gratis dan open source yang dapat dimanfaatkan untuk menyambungkan suatu alat ke internet salah satunya yaitu *SparkFun*[1,2]. *SparkFun* merupakan salah satu pendukung dalam membangun GUI (*Graphical User Interface*) yang digunakan untuk berbagai keperluan akses data dari server data lokal dan internet. Contoh alat yang dapat dipakai untuk memasukkan data ke server internet adalah arduino. Tidak hanya dapat mengakses ke server internet, arduino juga dapat dijadikan sebagai server lokal dalam mengakses suatu data, sehingga dapat dijadikan tempat sementara untuk mengambil maupun mengirim data yang menyebabkan akses data menjadi lebih cepat[3].

Banyak perangkat lunak pemrograman yang dapat digunakan untuk membuat browser sehingga teknik Web Programming menjadi cara yang realistis untuk membuat aplikasi *smartphone* yang fleksibel layaknya aplikasi yang bersifat open source sehingga bersifat multiguna ini. Aplikasi multiguna dapat dirancang memanfaatkan Web Browser sebagai media visual dari GUI yang dibuat. Aplikasi multiguna yang dimaksud dalam penelitian ini dirancang menggunakan *software Embarcadero Delphi XE8*[4]. *Embarcadero Delphi XE8* merupakan *software development* yang dipakai untuk membuat berbagai aplikasi[5]. Dalam *Embarcadero Delphi XE8* terdapat tools Web editor yang dapat digunakan untuk melengkapi aplikasi multiguna web browser yang dibuat, sehingga mendukung GUI, pengisian code HTML/Javascript, serta keperluan pengolahan maupun akses data dari server lokal dan internet.

Visualisasi GUI dirancang menggunakan code HTML/Javascript yang menggunakan library jQuery. Menggunakan library jQuery, code yang digunakan menjadi lebih singkat dan sederhana, sehingga tidak merumitkan pengguna yang berkebutuhan untuk mengubah visualisasi GUI yang dipakai.

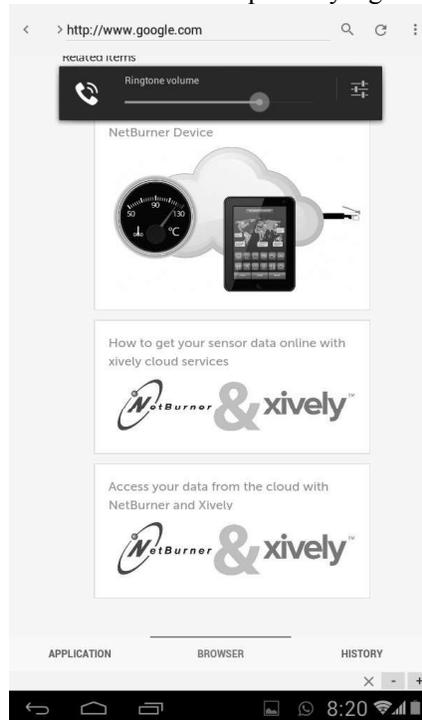
## 2. Pembahasan

Aplikasi multiguna yang dirancang berisikan komponen kontrol web browser yang dibuat menggunakan Embarcadero Delphi XE8 dengan API (*Application Programming Interface*) level 16, pada platform Android 4.1.2. Selain Web browser aplikasi yang dibangun dilengkapi dengan berbagai macam fitur atau komponen tambahan. Fitur tambahan untuk mendukung GUI diantaranya menampilkan code HTML, menyimpan code HTML ke dalam disk, membuka file HTML dari disk ke tool editor maupun langsung ditampilkan dalam layar, menjalankan code HTML, menuliskan history dari halaman aplikasi web yang pernah dibuka atau dikunjungi. Secara ringkas untuk menambahkan fitur-fitur di atas adalah sebagai berikut:

### 2.1. Pembuatan Aplikasi Web Browser

Pada form multi-device pada Delphi XE, beberapa komponen dapat ditambahkan antara lain komponen Tool Bar untuk menyimpan komponen edit, button, maupun combo edit dalam form. Kemudian dilakukan perubahan seperti align menjadi client. Selanjutnya menempatkan web browser pada form dan melakukan perubahan align menjadi client. Kelengkapan lain adalah komponen edit text untuk menuliskan alamat web/URL, 2 button untuk back dan forward. Button lainnya untuk melakukan reload dan search dan Button untuk menjalankan perintah operasi yang lain. Langkah terakhir adalah mengisi code untuk tiap-tiap komponen tersebut.

Kegiatan yang dilakukan di atas menghasilkan sebuah aplikasi yang disebut Web Browser. Gambar 1 Berikut merupakan tampilan dari Web Browser dari aplikasi yang dibangun.



Gambar 1. Aplikasi Web Browser.

### 2.2. Pengembangan Aplikasi Multiguna

Aplikasi tersebut di atas dapat ditambahkan pendukung yang berfungsi untuk mengakses data pada server yang dituju dan suatu visualisasi GUI. Komponen yang ditambahkan akan berfungsi sebagai wadah dalam menuliskan maupun mengedit HTML/Javascript, kemudian untuk menukar tampilan halaman web dengan teksnya. Beberapa tombol ditambahkan untuk mengoperasikan menyimpan file HTML, membuka file HTML ke dalam editor teks. Code-code HTML/JavaScript akan bekerja menjadi sebuah aplikasi tertentu, sehingga komponen kontrol berjenis comboedit ditempatkan untuk menuliskan nama file yang ingin disimpan dan memunculkan list file yang ada dalam penyimpanan, code ini merupakan bagian penting dari aplikasi multiguna yang dibangun. Seluruh komponen yang

disebutkan di atas ditempatkan dalam komponen treeview yang dapat di tampilkan secara on/off. Sedangkan untuk menjalankan code HTML/JavaScript yang berada pada penyimpanan disediakan sebuah tombol Run. Banyak code HTML/JavaScript dapat disimpan dan dijalankan melalui aplikasi yang dibangun ini sehingga secara kompak aplikasi ini berfungsi multiguna. Aplikasi ini juga dapat difungsikan menjadi sebuah web server yang dapat diakses menjadi server lokal untuk mengakses data yang telah diakses dari server data internet. Gambar 2 menampilkan desain aplikasi dan teks editor yang dibangun.



Gambar 2. Tampilan Aplikasi

*Alogaritma I: Aplikasi Inti ( Menggunakan Delphi)*

- i. Pilih file Aplikasi (code HTML/JavaScript) atau menulis aplikasi baru
- ii. Buka file di editor teks, operasi pengeditan dilakukan.
- iii. Jalankan di Browser (menurut algoritma II)
- iv. Kembali ke langkah i atau ke langkah v.
- v. Deteksi penekanan tombol untuk operasi pada Algoritma II atau III kalau tidak ke langkah vi
- vi. Keluar.

*Algoritma II: Run (Menjalankan code HTML/JavaScript)*

- i. Baca nama file (Aplikasi Browser)
- ii. Simpan file dalam disk
- iii. Buka file dari disk ke dalam Browser
- iv. Deteksi penekanan tombol sesuai langkah pada Algoritma I.

*Algoritma III: Save (Menyimpan Aplikasi dengan nama lain)*

- i. Baca teks dari editor dan nama file
- ii. Cek keberadaan nama file, jika ada yang sama ke langkah iii
- iii. Peringatan bahwa nama file telah ada, bila dipilih lanjut ke langkah iv bila tidak ke langkah v
- iv. Simpan file ke disk.
- v. Deteksi penekanan tombol sesuai langkah pada Algoritma I.

### 2.3. Komunikasi Data dengan Data Server *SparkFun*

Memasukkan data ke SparkFun membutuhkan data stream. *Data stream* dapat dibuat melalui situs *data.sparkfun.com*. Pada halaman utama klik *create*, maka akan muncul halaman “*create a data stream*”. Isikan kolom-kolom yang terdapat dalam halaman tersebut sesuai dengan keperluan data stream. Dimulai dengan kolom *title* untuk judul, *description* sebagai penjelasan dari *data stream*, lalu pilihan *show public stream list*. Kolom inti adalah pada bagian *fields*, bagian ini merupakan deretan kata yang akan mengisi kolom data dalam data stream. Kemudian *tags* untuk menyatakan data stream yang dibuat berhubungan dengan jenis stream seperti apa. Setelah selesai klik *save*. Kemudian akan diberikan *public* serta *private key* untuk mengakses data stream tersebut, baik untuk memasukkan data, mengakses data serta menampilkan data. Terdapat *public url* sebagai alamat yang bisa diakses untuk membuka data stream yang telah dibuat[6].

Untuk memasukkan data ke data stream dibutuhkan format, format tersebut telah diberikan oleh *SparkFun* saat selesai membuat data stream. Formatnya adalah: `http://data.sparkfun.com/input/[publicKey]?private_key=[privateKey]&[fields]=[value]&[fields]=[value]`. Contohnya adalah: `http://data.sparkfun.com/input/4J1DQ8vj3ETqL6xDqzYa?private_key=b51WpalxAkh25npg2dXx&cha1=19.87&cha2=26.37`[2]. Contoh tersebut untuk *fields* yang hanya memiliki dua *fields*, untuk lebih dari dua dapat meneruskan cara penulisan tersebut.

### 2.4 Aplikasi Grafik Akses Data Stream *SparkFun*

Membuat grafik membutuhkan pengaksesan data yang ada pada data stream. Untuk mengolah data tersebut agar bisa diplot menjadi grafik, diperlukan pemisahan-pemisahan data dari data keseluruhan. Data keseluruhan diperoleh dengan menggunakan function Send-Get sehingga diketahui output data keseluruhan pada data stream[7]. Setelah diketahui data keseluruhan barulah pemisahan data dilakukan sehingga hanya data yang diperlukan untuk diplot menjadi grafik yang tersisa. Pembacaan juga dapat dilakukan dengan memanfaatkan perintah Get AJAX. Library yang digunakan untuk menggambar grafik adalah jQuery library. Hasil grafik dalam browser ditunjukkan pada Gambar 5.

Berikut adalah source code inti dari keseluruhan aplikasi GUI membaca data dari server Spark Fun menggunakan ajax:

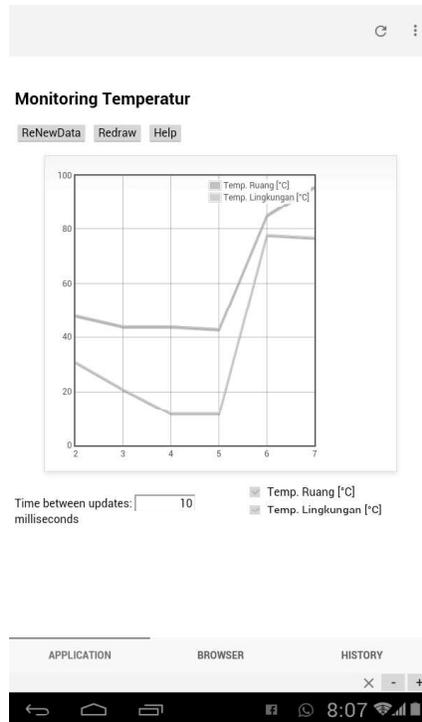
```
$.ajax({
  url: 'http://data.sparkfun.com/output/
4J1DQ8vj3ETqL6xDqzYa.json?timezone=asia/jakarta'+ '.json',
  data: {page: 1},
  dataType: 'jsonp',
}).done(function (results) {
  alert("Newdata Ready....");
  // loop through results and log temperature to the console
  var n=0;
  IsiDat=[]
  $.each(results, function (index, row) {
    n++;
    isiDat[0].push(row.number);
    isiDat[1].push(row.cha1);
    isiDat[2].push(row.cha2);
    isiDat[3].push(row.timestamp);
    jdatSF=n;
  });
  prosesData();
});
}
```

Gambar 3. Source code inti aplikasi GUI

dan Send-Get Http:

```
function loadXMLDoc() {
  StopP=true;
  var xmlhttp = new XMLHttpRequest();
  xmlhttp.onreadystatechange = function() {
    if (this.readyState == 4 && this.status == 200) {
      alert("Newdata Ready....");
      data0=this.responseText;
      res0=data0.split(",");
      jdatSF=res0.length;
      prosesData();
    }
  };
  xmlhttp.open("GET", "https://data.sparkfun.com/output/
4J1DQ8vj3ETqL6xDqzYa.json?
timezone=asia/jakarta"+" "+Math.random(), true);
  xmlhttp.send();
}
```

Gambar 4. Code Send-Get Http



Gambar 5. Tampilan Aplikasi Pembacaan Data ke Grafik.

Grafik ditampilkan menggunakan prinsip realtime sehingga dapat mengupdate data grafik secara otomatis saat data terus berubah. Dalam grafik di atas data ditampilkan secara bertahap yaitu 6 data. Pada kehidupan nyata aplikasi multiguna yang dibuat dapat diterapkan dalam berbagai bidang. Salah satu contohnya adalah dalam bidang industri pada bagian refrigerasi. Menggunakan arduino yang telah disinggung sebelumnya, aplikasi multiguna tersebut dapat memonitor temperatur pada *cold storage* yang dimanfaatkan sebagai penyimpanan baik untuk makanan maupun barang[8]. Pengaplikasian aplikasi multiguna ini dapat meningkatkan mutu penyimpanan *cold storage*, dan memberikan kemudahan bagi pelaku industri *refrigerasi*.

### 3. Simpulan

1. Perangkat lunak pemrograman *native* mendukung pembuatan komponen Web Browser sehingga memungkinkan sebuah aplikasi dapat difungsikan sebagai aplikasi multiguna.
2. Aplikasi multiguna berisi beberapa fungsi aplikasi khusus lain yang ditulis dalam code HTML/JavaScript.
3. Aplikasi multiguna yang dirancang mempunyai setidaknya tiga fungsi: sebagai web browser, menjalankan fungsi aplikasi sebagaimana yang dirancang dengan code HTML/JavaScript, dan sebagai web server lokal.
4. Aplikasi multiguna yang dirancang dibangun dapat menampilkan *Graphic User Interface* (GUI) pada *smartphone* sistem Android untuk menampilkan data *server data Spark Fun*.

### Daftar Pustaka

- [1]. <https://www.sparkfun.com/static/about>, diakses pada 5 Januari 2017
- [2]. <https://data.sparkfun.com/>, diakses pada 5 Januari 2017
- [3]. Pratikto, Faldian, Mughni Najmuddin, Raydha Zul Fitriani, "Automasi Bypass Valve pada Two-phase Refrigerant Mixture Dehumidifier Unit", in Proc. Thermofluid VIII 2016, pp 288, 10 November 2016.
- [4]. Pratikto, Susilawati, Bowo Yuli Prasetyo, Raydha Zul Fitriani, "Akuisisi Data Berbasis Smartphone untuk Unit Air Cooler and Dehumidifier", in Proc. SNTT 2016 Sekolah Vokasi UGM, 19 November 2016

- [5]. <https://www.embarcadero.com/products/rad-studio/start-for-free>, diakses pada 6 November 2016
- [6]. <https://learn.sparkfun.com/tutorials/pushing-data-to-datasparkfuncom/all>, diakses pada 18 November 2016
- [7]. [http://www.w3schools.com/xml/tryit.asp?filename=try\\_dom\\_xmlhttprequest\\_responsetext](http://www.w3schools.com/xml/tryit.asp?filename=try_dom_xmlhttprequest_responsetext), diakses pada 18 November 2016
- [8]. Pratikto, Aries P. Haryono, Ismail Wellid, Aditya Saputra, “Kontrol Temperatur *Cold Storage* Berbasis Arduino”, in Proc. IRWNS ke 7 2016, 28 Juli 2016