

Pendahuluan

Dahulu, membuat aplikasi mobile semacam android package (apk) boleh dibilang sulit dan tidak praktis. Seseorang butuh kemampuan bahasa pemrograman java yang boleh jadi bukanlah bahasa yang mudah karena mungkin menjadi bahasa baru baginya. Selain itu dibutuhkan file penunjang, yaitu sdk (software development kit) dengan ukuran yang relatif besar, mencapai 1,5 GBⁱ. Dengan kata lain, membutuhkan *resource* komputer yang besar.

Sekarang ini, membuat aplikasi mobile bukan lagi hal yang sulit dan cukup dengan *resource* yang relatif kecil. Seorang calon *developer* aplikasi mobile cukup memahami skrip HTML, dan Javascript. Skrip yang boleh dibilang relatif lebih mudah. Setidaknya lebih populer. Jika aplikasi bersifat client-sever, maka sang calon *developer* perlu memahami skrip PHP dan *database* seperti MySQL atau MariaDB.

Hal itu dimungkinkan dengan adanya *framework* bernama **PhoneGap**. PhoneGap

merupakan distribusi dari Apache Cordova, sebuah framework pengembangan aplikasi mobile yang memungkinkan penggunaan HTML5, CSS3 dan JavaScript.

*Apache Cordova is an open-source mobile development framework. It allows you to use standard web technologies - HTML5, CSS3, and JavaScript for cross-platform development.*ⁱⁱ

Dengan PhoneGap / Cordova, pengembang bukan hanya aplikasi mobile berbasis android, namun juga IOS, Windows mobile, BlackBerry, Ubuntu Phone, Firefox OS , LG Web OS, dan Fire OS.

Buku ini akan membahas langkah-langkah membangun aplikasi mobile berbasis android menggunakan *framework* PhoneGap / cordova melalui contoh proyek sederhana.

BAB I – KETERAMPILAN DASAR

Seperti diulas di atas, untuk membangun aplikasi mobile menggunakan *framework* PhoneGap / Cordova, pengguna perlu memahami skrip HTML, CSS dan JavaScript. Jika perlu, paham juga skrip PHP dan SQL. Dengan kata lain, untuk membuat aplikasi mobile menggunakan *framework* PhoneGap / Cordova, seseorang perlu bisa membuat halaman web terlebih dulu.

Karena halaman web yang dibuat nantinya tampil di perangkat mobile, maka halaman web yang dibuat harus bersifat *responsive*, mampu menyesuaikan tampilan pada ukuran perangkat (*device*) yang berbeda. Untuk membuat halaman web yang *responsive*, pengembang bisa memanfaatkan *framework* Bootstrap. Bootstrap sendiri merupakan *framework* yang menggabungkan HTML, CSS dan JavaScript untuk membangun *responsive web*ⁱⁱⁱ.

Bagian ini akan mengulas teori dan contoh penggunaan skrip HTML, CSS dan JavaScript

yang biasa digunakan dalam pengembangan aplikasi mobile.

HTML

Hypertext Markup Language atau HTML akan berfungsi sebagai wadah bagi konten aplikasi yang disajikan kepada pengguna. Meskipun HTML memiliki segudang tag yang bisa digunakan, di sini hanya akan dibahas *responsive web* dan tag-tag yang sering digunakan dalam sebuah aplikasi mobile.

Untuk membuat halaman web yang *responsive*, maka file html perlu menyematkan kelengkapan *framework* Bootstrap. Contoh halaman dasar web *responsive* sebagaimana dicontohkan di situs [www.w3schools^{iv}.com](http://www.w3schools.com) adalah sebagai berikut:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Bootstrap Example</title>
  <meta charset="utf-8">
  <meta name="viewport"
content="width=device-width, initial-
scale=1">
```

```

<link rel="stylesheet"
href="https://maxcdn.bootstrapcdn.com/bootst
rap/3.3.7/css/bootstrap.min.css">
  <script
src="https://ajax.googleapis.com/ajax/libs/
jquery/3.1.1/jquery.min.js"></script>
  <script
src="https://maxcdn.bootstrapcdn.com/bootst
rap/3.3.7/js/bootstrap.min.js"></script>
</head>
<body>

<div class="container-fluid">
  <h1>My First Bootstrap Page</h1>
  <p>This is some text.</p>
</div>

</body>
</html>

```

Tag-tag yang sering digunakan antara lain:

Division

Tag <div> digunakan untuk mengelompokkan elemen-elemen dokumen HTML dan memformatnya melalui CSS^v. Contoh penggunaan tag <div>.

```

<div style="background-color:navy;
color:silver;">
  <h4>Berita Hari Ini</h4>

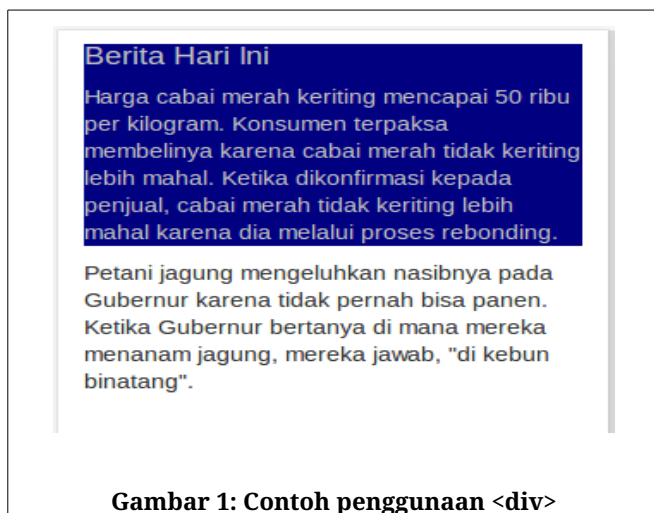
```

<p>Harga cabai merah keriting mencapai 50 ribu per kilogram. Konsumen terpaksa membelinya karena cabai merah tidak keriting lebih mahal. Ketika dikonfirmasi kepada penjual, cabai merah tidak keriting lebih mahal karena dia melalui proses rebonding.</p>

</div>

<p>Petani jagung mengeluhkan nasibnya pada Gubernur karena tidak pernah bisa panen. Ketika Gubernur bertanya di mana mereka menanam jagung, mereka jawab, "di kebun binatang".</p>

Hasilnya menjadi seperti berikut:



Walau sama-sama tag paragraf / <p>, namun yang berada di dalam elemen <div> berwarna latar belakang biru laut.

Heading

Heading umumnya digunakan untuk menampilkan judul. Oleh karenanya ukuran hurufnya lebih besar. Contoh:

```
<h1>Heading 1</h1>  
<h2>Heading 2</h2>  
<h3>Heading 3</h3>  
<h4>Heading 4</h4>
```

Hasilnya seperti berikut:

Heading 1

Heading 2

Heading 3

Heading 4

Gambar 2: Contoh penggunaan heading

Paragraph

Paragraph digunakan untuk menandai sekumpulan kalimat. Biasanya diterapkan pada artikel atau uraian suatu tema. Contoh penggunaan paragraph bisa dilihat di gambar 1.

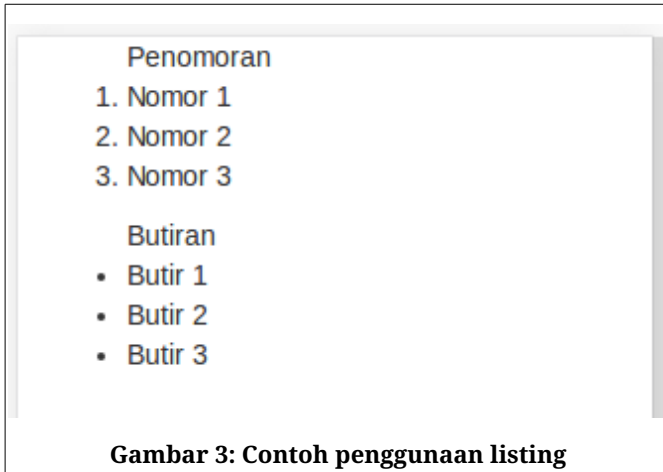
Listing

Listing terdiri dari penomoran dan butiran. Contohnya sebagai berikut:

```
<ol>Penomoran
  <li>Nomor 1</li>
  <li>Nomor 2</li>
  <li>Nomor 3</li>
</ol>

<ul>Butiran
  <li>Butir 1</li>
  <li>Butir 2</li>
  <li>Butir 3</li>
</ul>
```


Hasilnya seperti berikut:



Namun dalam penerapan di pengembangan aplikasi mobile, listing dimodifikasi menggunakan bootstrap agar isinya bisa terdiri dari beragam tag. Contoh:

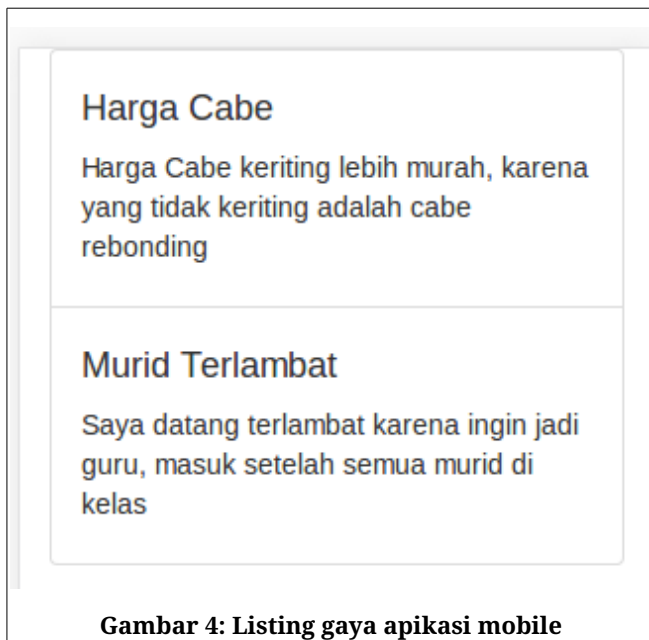
```
<ul class='list-group'>
  <li class="list-group-item">
    <h4>Harga Cabe</h4>
    <p>Harga Cabe keriting lebih murah,
karena yang tidak keriting adalah cabe
rebonding</p>
  </li>

  <li class="list-group-item">
```

```
<h4>Murid Terlambat</h4>
  <p>Saya datang terlambat karena
ingin jadi guru, masuk setelah semua murid
di kelas</p>
</li>

</ul>
```

Class `list-group` dan `list-group-item`^{vi} adalah class bawaan dari bootstrap. Dengan class-class tersebut hasilnya menjadi sebagai berikut:



Gambar 4: Listing gaya aplikasi mobile