

# FRAMEWORK PHONEGAP SEBAGAI TEKNOLOGI CROSS-PLATFORM MOBILE DEVELOPMENT: STUDI KASUS KAMUS TUMBUHAN

Ari Muzakir<sup>1</sup>

Universitas Bina Darma<sup>1</sup>

Jl. A Yani No.03 Palembang Sumatera Selatan, (0711)515582

e-mail: arimuzakir@binadarma.ac.id<sup>1</sup>

**Abstrak**— Mobile based application is now very quick to deploy. This is in line with the availability of smartphone devices available in the market and supported by the prices are getting cheaper. Different types of platforms available on smartphones such as Android, iOS, Windows Phone (WP) and the Blackberry, forcing application developers to be able to adjust to the needs of users with a variety of smartphone platforms. To overcome this, the emerging framework that provides solutions phonegap cross-platform issues. Phonegap itself is a hybrid application development platform that allows users to build mobile apps using HTML, CSS, and JavaScript (JS). Experiments were carried out in this research is to build a dictionary application of scientific plant names into Indonesian. This trial resulted in mobile apps on android and windows mobile platform that can be build without a hitch. While on iOS development constrained in license certificate.

**Keyword**— *Framework Phonegap, Cross-platform, hybrid mobile development, HTML5 applications, hybrid programming.*

**Intisari** - Aplikasi berbasis mobile saat ini sangat cepat dalam pengembangannya. Hal ini seiring dengan tersedianya device smartphone yang ada di pasaran serta ditunjang dengan harga yang semakin murah. Berbagai jenis platform yang tersedia pada smartphone seperti Android, IOS, Windows Phone (WP), Blackberry memaksa para pengembang aplikasi untuk mampu menyesuaikan dengan kebutuhan pengguna dengan berbagai platform smartphone. Untuk mengatasi hal ini, muncul framework phonegap yang memberi solusi permasalahan cross-platform. Phonegap sendiri merupakan hybrid application development platform yang memungkinkan pengguna untuk membangun mobile apps menggunakan HTML, CSS, dan Javascript (JS). Ujicoba yang dilakukan dalam penelitian ini adalah membangun aplikasi kamus ilmiah nama tumbuhan ke dalam bahasa Indonesia. Ujicoba ini menghasilkan aplikasi mobile pada platform android dan windows mobile yang dapat di-build tanpa hambatan. Sedangkan pada IOS terkendala pada license certificate development.

**Kata Kunci:** Framework Phonegap, Cross-platform, hybrid mobile development, aplikasi HTML5, hybrid programming.

## 1. PENDAHULUAN

Sejak 2010, jumlah perangkat mobile telah meningkat lebih dari 15% per kuartal, yang lebih menekankan kebutuhan

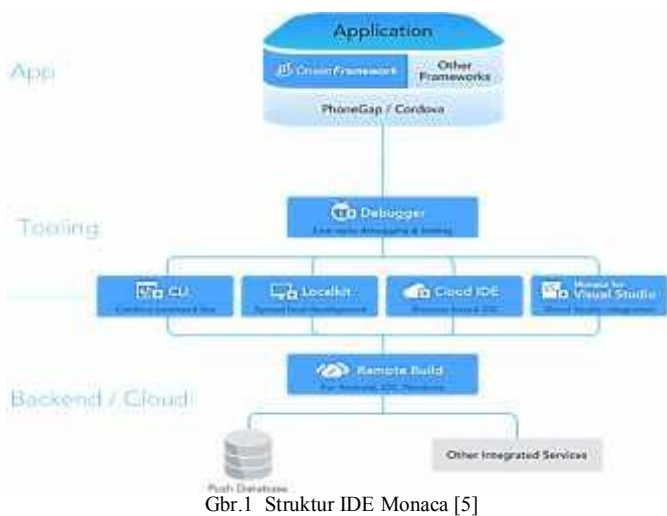
untuk kecukupan perusahaan, instansi pemerintah dan perguruan tinggi untuk realitas baru ini. Bagi perusahaan, kurangnya pengetahuan teknis telah hambatan untuk tidak menerapkan ide-ide yang baik karena biaya yang diperlukan untuk pelatihan [7].

Perkembangan ilmu pengetahuan dan teknologi kini semakin maju. Kemajuan teknologi dan informasi dapat dilihat dengan semakin banyaknya penggunaan smartphone sebagai alat bantu yang mutakhir, yang bertujuan untuk mempermudah pekerjaan manusia, sehingga waktu yang digunakan semakin cepat, dan mudah. [3] Perkembangan smartphone telah memicu pola pikir masyarakat dalam memandang smartphone sebagai alat yang penting untuk kehidupan saat ini. Kemudahan penggunaan smartphone sudah menjadi keharusan bagi perancang maupun bagi para pembuat aplikasi atau software (programmer). Pengembangan aplikasi mobile harus ingat bahwa dengan tersedianya berbagai platform smartphone yang berbeda-beda dan membangun aplikasi untuk smartphone tersebut tentunya akan memakan waktu dan biaya yang mahal. Selain itu juga pada proses pengembangan tentunya juga akan menemukan berbagai bentuk permasalahan dikarenakan beda platform pasti beda bahasa pemrograman yang digunakan. Sebuah arsitektur multiplatform akan menjadi solusi untuk membuat tugas yang sulit ini menjadi sesuatu yang jauh lebih terjangkau, dan dengan kemungkinan pengembangan menggunakan metode web atau pengembangan aplikasi hybrid [8].

Teknologi phonegap membuktikan menjadi salah satu solusi yang paling populer untuk permasalahan cross-platform, hal ini memungkinkan pengguna untuk dapat dengan cepat dan mudah membangun aplikasi yang dijalankan pada beberapa platform yang berbeda [2]. Phonegap merupakan hybrid apps development platform yang mendukung beberapa platform seperti iOS, Android, Blackberry, Windows Phone (WP). Phonegap sendiri merupakan framework yang memungkinkan pengguna untuk mengembangkan aplikasi mobile menggunakan HTML, CSS, dan JavaScript (JS) [1,9]. Setelah munculnya teknologi ini di dunia mobile, ada banyak kerangka kerja dan kemudian pembagian solusi mobile dalam tiga kategori: asli, hybrid dan web apps [10].

Secara umum sistem yang akan dibangun dalam penelitian ini adalah melakukan rancang bangun aplikasi kamus ilmiah nama tumbuhan. Dalam penelitian ini, ujicoba yang dilakukan menggunakan bahasa pemrograman HTML5, CSS, dan Javascript yang mana sudah terpaket dalam framework Phonegap. Sedangkan untuk backend menggunakan pemrograman PHP. Tool yang digunakan dalam penyelesaian aplikasi ini memanfaatkan integrated development environment (IDE) secara online dari Monaca yang merupakan suatu tools dan services untuk membangun suatu sistem mobile hybrid application dengan menggunakan HTML5 dan Phonegap. Monaca sangat memungkinkan dalam membangun aplikasi lintas platform seperti Android, iOS, Windows Phone dan lain sebagainya karena dibangun diatas aplikasi open source Cordova. Kerangka (framework) untuk pengembangan aplikasi hybrid dipisahkan menjadi dua jenis, mereka yang bertanggung jawab untuk antarmuka (UI - user interface), dan yang bertanggung jawab untuk mengemas aplikasi menjadi platform yang berbeda dan memungkinkan untuk mengakses fitur telepon [11].

Adapun untuk membangun interface pada aplikasi ini menggunakan framework Onsen UI. Framework Onsen UI sendiri tersusun dari HTML, CSS, dan Javascript (AngularJs) layaknya framework Bootstrap yang biasa digunakan dalam membangun sistem berbasis web. Adapun struktur dari Monaca dapat dilihat pada gambar 1 berikut [5].



Gbr.1 Struktur IDE Monaca [5]

Selain itu juga, smartphone sangat membantu dan sudah merambah ke dunia pendidikan diantaranya dalam bidang ilmu biologi. Dijelaskan bahwa ilmu biologi memiliki karakteristik khusus, yang berbeda dengan ilmu lain dalam hal objek, persoalan, dan metodenya. Berdasarkan keilmuan menurut BSCS (Biological Science Curriculum Study), biologi mempunyai obyek berupa kerajaan (kingdom): plantae (dunia tumbuhan) animalia (dunia hewan), dan Protista [ 4 ]. Nama nama hewan (animalia) dan tumbuhan (plantae) yang digunakan dalam pembelajaran biologi di sekolah, Sekolah

Menengah Pertama (SMP), dan Sekolah Menengah Atas (SMA), dan di Perguruan Tinggi banyak disajikan dalam bahasa Latin. Namun, banyak penggunaan tidak didampingi nama dalam bahasa Indonesia, sehingga bila siswa atau mahasiswa ingin mengetahui nama tumbuhan yang telah diketahui dan ingin menerjemahkan ke dalam nama ilmiah mengalami kesulitan karena begitu banyak nama atau istilah yang harus dicari. Sehingga berdasarkan uraian tersebut, maka diambil kesimpulan bahwa diperlukan aplikasi yang bisa memberikan informasi yang lengkap tidak hanya fokus pada nama latin latin saja melainkan meliputi filum, class, ordo, dan familia serta penjelasan yang berubungan dengan informasi tumbuhan tersebut. Untuk itu dalam penelitian ini membahas mengenai rancang bangun aplikasi penerjemah nama tumbuhan ke dalam bahasa ilmiah menggunakan framework phonegap berbasis smartphone dengan menggunakan bahasa pemrograman HTML5 dan PHONEGAP sebagai framework dalam penerapannya.

## 2. METODE PENELITIAN

Metode penelitian yang digunakan adalah metode deskriptif kualitatif yang disesuaikan dengan metode pengembangan perangkat lunak. Dalam penerapannya, proses analisis dilakukan dengan melihat objek pengguna yaitu mengenai pemahaman mengenai istilah nama ilmiah tumbuhan. Dari hasil analisa tersebut nantinya akan dipecahkan menggunakan aplikasi kamus istilah nama ilmiah tumbuhan.

Adapun model yang digunakan dalam proses pengembangan kamus ilmiah tumbuhan ini menggunakan model waterfall. Tahapan-tahapan yang ada dalam model waterfall ini yaitu analisis, desain, pengkodean, dan pengujian [6].

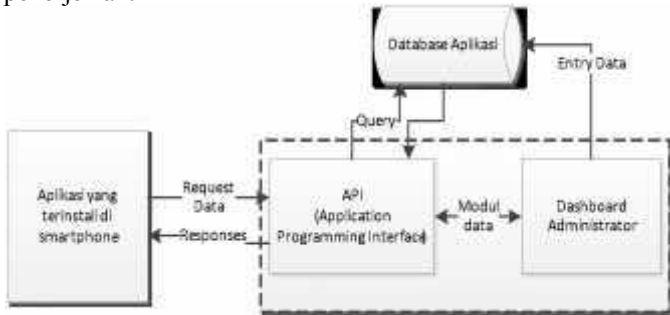
### 2.1. Analisis Sistem

Secara umum sistem yang akan dibangun dalam penelitian ini adalah melakukan rancang bangun aplikasi penerjemah nama tumbuhan ke dalam bahasa ilmiah dengan menggunakan framework phonegap berbasis smartphone. Dalam penelitian ini menggunakan bahasa pemrograman HTML5, CSS, dan Javascript yang mana sudah terpaket dalam framework Phonegap. Adapun tool yang digunakan dalam penyelesaian aplikasi ini memanfaatkan integrated development environment (IDE) secara online dari Monaca yang merupakan suatu tools dan services untuk membangun suatu sistem mobile hybrid application dengan menggunakan HTML5 dan Phonegap. Monaca sangat memungkinkan dalam membangun aplikasi lintas platform seperti Android, iOS, Windows Phone dan lain sebagainya karena dibangun diatas aplikasi open source Cordova.

Adapun untuk membangun interface pada aplikasi ini menggunakan framework Onsen UI. Framework Onsen UI sendiri tersusun dari HTML, CSS, dan Javascript (AngularJs) layaknya framework bootstrap yang biasa digunakan dalam membangun sistem berbasis web.

Pada implementasinya nanti, pengguna akan mengakses melalui sebuah aplikasi penerjemah yang terpasang di

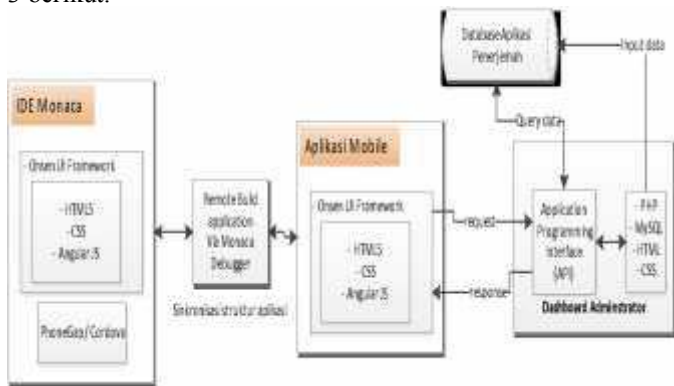
smartphone. Seluruh proses dalam aplikasi ini akan selalu terhubung menggunakan jaringan internet untuk dapat menggunakannya. Pada saat terjadi request dan response data akan melibatkan modul application programming interface (API), yang mana didalam modul API tersebut terdapat suatu fungsi query ke database. Selain itu juga terdapat sebuah dashboard untuk login sebagai administrator untuk memperbarui seluruh ini konten aplikasi penerjemah. Berikut di ilustrasikan dalam gambar 2 mengenai proses dari aplikasi penerjemah.



Gbr.2 Proses dalam aplikasi penerjemah menggunakan *framework* phonegap

2.2. Perancangan Sistem

Perancangan (*design*) merupakan tahapan ke tiga dari model *linear sequential* dimana pada tahap ini proses desain menerjemahkan kebutuhan kedalam ke dalam sebuah representasi perangkat lunak yang dapat diperkirakan demi kualitas sebelum dimulai generasi kode. Aplikasi yang dibangun memiliki arsitektur seperti ditunjukkan pada gambar 3 berikut.



Gbr.3 Alur arsitektur phonegap aplikasi penerjemah nama tumbuhan

Pada gambar 3 tersebut dapat diperjelas melalui beberapa alur berikut:

1. Pada tahap awal membangun phonegap *framework* ini menggunakan IDE online dari Monaca. IDE Monaca ini diakses secara *online* artinya ketika memulai membuat aplikasi phonegap ini harus terkoneksi dengan *internet*. Monaca dipilih karena kemudahan dalam membuat aplikasi disertai dengan fitur *build* secara langsung ke beberapa ekstensi *file* di *smartphone*. Tahap awal ini menggunakan *framework* onsen UI dalam membuat *interface*. Onsen UI sendiri memiliki kelengkapan fitur

yang dibutuhkan dalam membangun aplikasi berbasis phonegap seperti HTML5, CSS, dan *Javascript* (Angula JS).

2. Tahap kedua adalah melakukan sinkronisasi ke aplikasi IDE Monaca ke Monaca *Debugger* (terinstall di *smartphone*) sebagai *emulator preview*.
3. Untuk dapat terintegrasi dengan data yang ada di database, maka dibangun dashboard untuk administrator. Dimana data akan selalu tersinkronisasi melalui API yang dibangun dan di letakkan pada suatu server (dalam hal ini nantinya akan dipaketkan dengan aplikasi dashboard administrator). Proses sinkronisasi dan pengambilan data dari aplikasi pengguna ke database menggunakan struktur javascript object notation (JSON).

3. HASIL DAN PEMBAHASAN

Setelah melalui beberapa analisis kebutuhan sistem serta perancangan desain yang berkaitan dengan aplikasi penerjemah nama ilmiah tumbuhan, maka didapatkan hasil berupa aplikasi sistem. Hasil yang diperoleh ada 2 jenis yaitu sistem berbasis web sebagai tempat input data kosakata nama tumbuhan dan sistem berbasis *smartphone*.

Untuk ujicoba pertama kali dilakukan pada menggunakan *emulator* IDE Monaca secara langsung. Selanjutnya untuk proses pengujian fungsionalnya menggunakan Monaca Debugger. Berikut gambar 4 memperlihatkan mengenai tampilan dari emulator IDE Monaca yang akses secara *online* melalui alamat <http://monaca.io>.



Gbr.4 Hasil Emulator pada IDE Monaca secara *online*

Pengujian yang dilakukan menggunakan metode *blackbox* didapatkan hasil seperti tabel 1 berikut.

TABEL I  
HASIL UJICOBA MENGGUNAKAN METODE BLACKBOX

Kasus dan Hasil Uji (Data Normal)			
Data Masukan	Yang diharapkan	Pengamatan	Kesimpulan



Halaman Utama Aplikasi Mobile (fitur pencarian)	Fitur pencarian berdasarkan nama ilmiah (Indonesia atau latin) sesuai dengan yang dicari	Menampilkan hasil pencarian secara mendetil	[ x ] diterima [ ] ditolak
<b>Kasus dan Hasil Uji (Data Salah)</b>			
<b>Data Masukan</b>	<b>Yang diharapkan</b>	<b>Pengamatan</b>	<b>Kesimpulan</b>
Nama tumbuhan (baik Indonesia atau latin) tidak terdapat pada database	Tidak menampilkan data apapun pada tampilan layar	Hasil pencarian tidak ada	[ x ] diterima [ ] ditolak

				pada windows 8.x
Konfigurasi App	mudah	Mudah jika memiliki sertifikat	mudah	Jenis konfigurasi antara lain: <i>splash screen, app name, version device, private key</i>
Biaya Publish App to Store	USD 25 (dengan kartu kredit)	USD 99 / tahun (dengan kartu kredit)	USD 99 / tahun (dengan kartu kredit)*	Untuk windows phone gratis jika teregistrasi di DreamSpark Student
File Installer for testing	Tersedia file .apk	tidak	Tersedia file .appx	Untuk IOS langsung di <i>publish</i> ke app store

Hasil pencarian kosakata dapat dilihat secara detil beserta penjelasan lengkap dan gambar dari nama tumbuhan. Data-data tersebut disinkronisasi secara *online* atau langsung melalui API ke *database* yang telah dibuat. Hasil detil dari aplikasi dapat di lihat pada gambar 5 berikut. Selain itu juga dari menu kosakata pada aplikasi akan menampilkan seluruh daftar kosakata.



Gbr.5 Hasil detil pencarian menggunakan *emulator* pada Monaca berbasis android

Pengujian yang dilakukan aplikasi baik secara online menggunakan *emulator* IDE Monaca maupun Monaca pada android berjalan secara normal. Untuk itu dapat dilakukan proses build aplikasi ke dalam *smartphone* berbasis *mobile*. Pengujian *build* yang dilakukan menunjukkan bahwa dengan menggunakan pemrograman HTML5, CSS, dan *Javascript* dapat menghasilkan suatu aplikasi lintas *platform* (*cross platform*). Hasil kesimpulan pada saat *build* dapat dilihat pada tabel 2 berikut.

TABEL 2  
HASIL BUILD APLIKASI HTML5 KE SMARTPHONE

Jenis Pengujian	Jenis Smartphone			Keterangan Hasil
	Android	IOS	Windows Phone	
Build App file secara free	berhasil	Belum berhasil	berhasil	IOS membutuhkan generate private key dan juga sertifikat development langsung dari Apple yang membutuhkan biaya sekitar 99 dolar - Pada Windows Phone hanya dapat berjalan

Pada pembahasan di penelitian ini terdapat dua buah proyek yang dibuat yaitu proyek kamus yang terinstal di *smartphone* dan proyek kamus sebagai *dashboard* untuk administrator dalam memanajemen kosakata. Pada proyek kamus untuk manajemen kosakata disisipkan sebuah *file application programming interface* (API) yang dibangun menggunakan program PHP. Pada *file* API ini berisi modul-modul yang digunakan untuk *query* data secara langsung antara aplikasi *smartphone* dengan dengan *database*. Peran *framework* Phonegap disini adalah sebagai kerangka kerja yang berisi HTML5, CSS, dan *Javascript*. HTML5, CSS, dan *Javascript* sendiri digunakan sebagai template pada saat *development* aplikasi penerjemah atau kamus yang sudah dikemas dalam *framework* Onsen UI. Berikut pada gambar 6 potongan *coding* struktur *index* dari proyek kamus.



Gbr.6 Potongan *coding* program pada struktur *index*

Pada gambar tersebut dapat dijelaskan bahwa struktur program tersebut menggunakan *template* dari *framework* Onsen UI. Onsen UI sendiri menggunakan struktur HTML, CSS, dan Angular JS. Hal tersebut dapat dilihat pada baris ke 1 sampai 22. Pada baris ke 20 dan 21 menjelaskan mengenai modul yang digunakan dalam membangun aplikasi Phonegap, kemudian mendefinisikan alamat API dari aplikasi untuk sinkronisasi atau *request data*. Struktur dari API dari aplikasi kamus dapat dilihat dari potongan skrip program berikut.

