

## Pengembangan Aplikasi Hybrid Menggunakan Ionic 2 Framework dan Angular 2

Paula Dewanti<sup>1)</sup>, Putu Adi Guna Permana<sup>2)</sup>

STMIK STIKOM Bali/Sistem Komputer  
Jl. Raya Puputan No.86, Renon – Denpasar, (0361)244446  
e-mail: pdewanti@gmail.com<sup>1)</sup>, adiguna.permana@gmail.com<sup>2)</sup>

### Abstrak

Pengembangan perangkat lunak di Indonesia berkembang sangat pesat, terutama perangkat lunak berbasis *mobile*. Banyak aplikasi dikembangkan pada berbagai platform, seperti Android, iOS dan Windows Phone. Mengembangkan aplikasi *cross platform*, untuk membentuk aplikasi pada berbagai macam platform khusus, selain tidak efisien, juga sangat merepotkan dikarenakan komplikasi penggunaan Java, Objective-C atau Swift. Bagaimana sebuah software dapat dijalankan di *mobile operating system* yang berbeda, merupakan salah satu fokus pengembangan saat ini. Ionic Framework merupakan salah satu solusi yang dapat dipergunakan. Penggunaan Ionic Framework memungkinkan pengembang membuat aplikasi untuk beberapa platform dengan hanya menggunakan bahasa pengembangan web untuk membuat sebuah aplikasi *mobile*. Framework Ionic 2 berbasis *mobile* merupakan kerangka kerja dari sebuah software yang dapat dijalankan di *mobile operating system* yang berbeda seperti Android, iOS dan Windows Phone. Konsep Ionic 2 ini adalah membuat aplikasi yang dapat dijalankan pada OS yang berbeda. Dengan adanya konsep ini maka perusahaan dapat memangkas biaya pembuatan aplikasi di OS berbasis *mobile*, yang berbeda-beda. Hasil dari penelitian ini adalah membuat aplikasi, dengan mengimplementasikan Framework Ionic 2, yang dapat dijalankan pada sistem operasi yang berbeda.

**Kata kunci:** Android, Angular 2, Hybrid Application, Ionic 2 Framework, Windows Phone

### 1. Pendahuluan

Pengembangan perangkat lunak di Indonesia berkembang sangat pesat, terutama perangkat lunak berbasis *mobile*. Perusahaan berskala kecil, menengah maupun besar, mengembangkan perangkat lunak berbasis *mobile* untuk memudahkan akses bagi para pengguna tanpa dibatasi oleh waktu dan lokasi, dan tanpa harus membuka PC maupun Notebook. Framework Ionic 2 berbasis *mobile* merupakan kerangka kerja (*framework*) dari sebuah *software* yang bisa dijalankan di *mobile operating system* yang berbeda seperti Android, iOS dan Windows Phone.

Android adalah sebuah sistem operasi pada handphone yang bersifat terbuka dan berbasis pada sistem operasi Linux. Android bisa digunakan oleh setiap orang yang ingin menggunakannya pada perangkat mereka. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri yang digunakan untuk bermacam peranti bergerak.

Konsep Ionic 2 ini adalah membuat aplikasi yang dapat dijalankan pada sistem operasi yang berbeda. Ionic Framework adalah SDK (*Software Development Kit*) *open source* yang memungkinkan pengembang untuk membangun aplikasi *mobile* berkualitas tinggi menggunakan teknologi web yang familiar (HTML, CSS, dan JavaScript). Ionic difokuskan terutama pada tampilan dan nuansa, atau interaksi user UI (*User Interface*), dari sebuah aplikasi.

Angular 2 adalah iterasi selanjutnya dari AngularJS Framework. Angular 2 Framework menjanjikan kinerja yang lebih baik dan menggunakan bahasa pemrograman TypeScript untuk pemrograman yang aman. Sementara Apache Cordova memungkinkan pengembang membuat aplikasi *mobile* menggunakan JavaScript, CSS dan HTML daripada menggunakan platform antarmuka pemrograman aplikasi spesifik (APIs) seperti yang ada di iOS, Android atau Windows Phone. Ionic Framework adalah kit pengembangan perangkat lunak *open source* yang digunakan di untuk membuat aplikasi *mobile hybrid*. Dibangun dengan menggunakan AngularJS dan Apache Cordova. Ionic menyediakan pengguna dengan semua komponen, peralatan dan fungsi yang digunakan dalam pengembangan seluler asli - perangkat pengembangan perangkat lunak (SDKs). Pengembang bisa mendesain Aplikasi mereka menggunakan alat dan kode contoh yang disediakan oleh dokumentasi kerangka kerja dan bantuan situs web. Instalasi itu membutuhkan NodeJS dan npm yaitu pengelola paket bawaan untuk NodeJS, terlepas dari Windows, lingkungan Linux atau Macintosh. Ionic adalah platform yang jauh lebih baik untuk digunakan dalam rangka merancang dan membangun aplikasi, karena aplikasi

asli ada platform spesifik, memiliki alat pengembangan masing-masing, sangat memakan waktu dan mahal juga. Ionic on sisi lain adalah platform independen, menggunakan HTML, CSS dan JavaScript. Proses pembangunannya cukup cepat dan pengembang memiliki akses langsung API dengan Cordova.

Penelitian dilakukan untuk mengetahui bagaimana Pengembangan Aplikasi Hybrid menggunakan Ionic 2 Framework dan Angular 2, yang dapat dijalankan pada sistem operasi yang berbeda, untuk menghasilkan efisiensi didalam proses pembuatan perangkat lunak berbasis *mobile*.

## 2. Metode Penelitian

### 2.1. Instalasi dan Pengaturan

Prosedur instalasi dari Ionic dan Cordova cukup sederhana. Sebelum installasi Ionic 2 dan Cordova diharapkan agar melakukan installasi Node JS terlebih dahulu dimana source bisa di download pada nodejs.org. Untuk melakukan instalasi global, perintah berikut dapat di masukan di terminal yang telah menginstal paket yang dibutuhkan.

Pemasangan global: **npm install -g ionic cordova**

Setelah aplikasi dasar Ionic 2 sudah ter-*install*, silahkan gunakan command berikut untuk pengecekan :

**ionic -v**  
**cordova-v**

Untuk membangun aplikasi baru di Ionic 2, **ionic serve** dapat digunakan di terminal setelah melakukan setting direktori yang sesuai dimana proyek tersebut tersimpan. Hasil dari perintah tersebut akan menghasilkan tampilan aplikasi pada *browser default* pada komputer. Jika ingin menampilkan hasil aplikasi dengan berbagai versi pada browser bisa menggunakan **ionic serve -cls** , maka akan tampil 3 versi tampilan yaitu Android, Windows dan iOS.

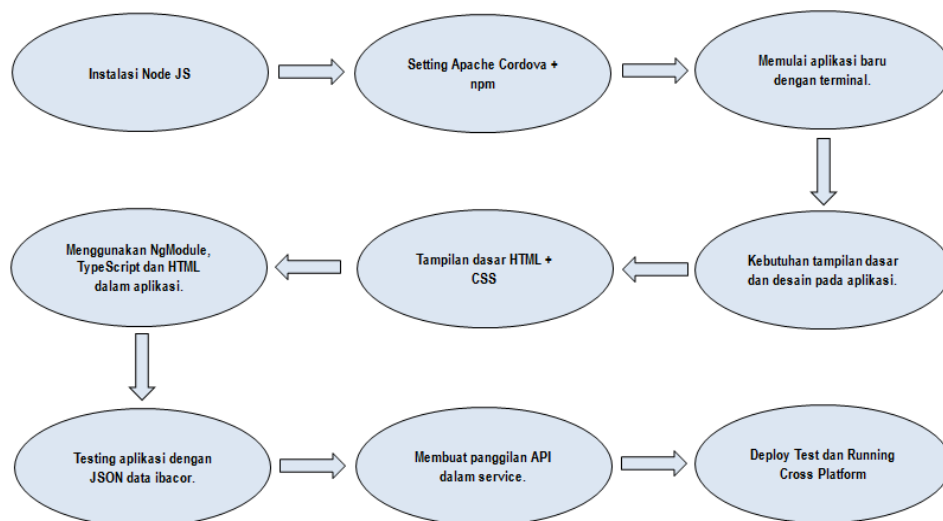
Untuk menjalankan aplikasi pada emulator/*device phone* harus menambahkan platform pada project aplikasi dengan cara : **ionic platform add [android/ios]**, pada terminal. Ketika ingin menjalankan aplikasi pada emulator bisa menggunakan *command* : **ionic [run/emulate] [android/ios]**

### 2.2. Spesifikasi Perangkat Lunak

Untuk membangun aplikasi, beberapa *tools* dan *software* dipergunakan. Aplikasi dibangun dan diuji pada *platform* Windows menggunakan *command line interface*. Salah satunya adalah penggunaan NodeJS Versi 4.4.0. Untuk menuliskan TypeScript dan CSS Script, editor teks yang digunakan adalah Visual Studio Code. *Web Browser* yang digunakan untuk *deployment* dan *debugging* adalah Google Chrome versi 49.0.2623.87.

### 2.3. Adaptasi Metodologi Pengembangan Aplikasi Hybrid

Untuk membangun aplikasi dari awal, sketsa konsep dasar digambarkan seperti Gambar 1. dibawah ini. Sketsa konsep menyoroti berbagai aspek yang dibutuhkan dalam rangka merancang, membangun, menguji, debug dan menjalankan aplikasi.



Gambar 1. Adaptasi Metodologi Pengembangan Aplikasi Hybrid

---

### 3. Hasil dan Pembahasan

#### 3.1. Perancangan dan Implementasi Kode

Aspek terpenting untuk melakukan pengkodean pada aplikasi *mobile* adalah pemahaman yang mendalam tentang Angular 2 Framework. Terdapat berbagai aspek yang perlu dipahami untuk dapat memahami pengembangan aplikasi menggunakan Ionic 2 dan Angular 2 seperti *NgModule*, *Component*, dan HTML Angular 2 dibangun untuk memberikan fungsionalitas pada aplikasi. Angular 2 juga memungkinkan untuk menentukan *Component* kita sendiri. *NgModule* pada Angular 2 digunakan untuk mengendalikan data dalam aplikasi dimana digunakan objek TypeScript dasar. *Controller* setiap *page* HTML dikendalikan pada *code* TypeScript. Nama *class* yang ada didalam TypeScript masing – masing *controller* akan di daftarkan didalam **app.module.ts**. Tampilan beberapa *code*:

Pada *Controller Page*:

```
xport class HelloIonicPage {
  dataZodiak = {tgl_lahir : "",
    nama: ""};
  itemsZodiak: any;
  tgl_lahir;
  constructor(public navCtrl: NavController, private zodiakService:ZodiakService, public
  alertCtrl: AlertController) {
  }

  searchZodiak(){}
  .....
}
```

Pada *app.module.ts*:

```
@NgModule({
  declarations: [
    MyApp,
    HelloIonicPage,
    ItemDetailsPage,
    ListPage
  ],
  imports: [
    IonicModule.forRoot(MyApp)
  ],
  bootstrap: [IonicApp],
  entryComponents: [
    MyApp,
    HelloIonicPage,
    ItemDetailsPage,
    ListPage
  ],
  providers: [
    StatusBar,
    SplashScreen,
    {provide: ErrorHandler, useClass: IonicErrorHandler}
  ]
})
```

HTML pada dasarnya digunakan untuk mendefinisikan interface tanpa tampilan sebuah antarmuka, pada Ionic 2 ini bisa menggunakan *tag* bawaan Ionic ataupun HTML dasar, contoh *code*, seperti:

```
<ion-menu [content]="content">
  <ion-content class="has-header mymenu">
    <ion-item class="menu-header">
      <div class="avatar">
        <br/>
        <div class="app-icon"></div>
      </div>
    </ion-item>
  </ion-content>
</ion-menu>
```

```

<br/>
<h2>My Zodiak</h2>
</div>
</ion-item>
<br/>
<ion-list class="myitem">
  <button menuClose ion-item *ngFor="let p of pages" (click)="openPage(p)">
    <ion-icon name="{{p.icon}}" item-left></ion-icon>
    {{p.title}}
  </button>
</ion-list>
</ion-content>
<ion-footer>
  <ion-buttons class="mybutton">
    <button ion-button outline (click)="exitApp()" >
      <ion-icon name="log-out" item-left></ion-icon>
      Keluar
    </button>
  </ion-buttons>
</ion-footer>
</ion-menu>

<!-- Disable swipe-to-go-back because it's poor UX to combine STGB with side menus -->
<ion-nav [root]="rootPage" #content swipeBackEnabled="false"></ion-nav>

```

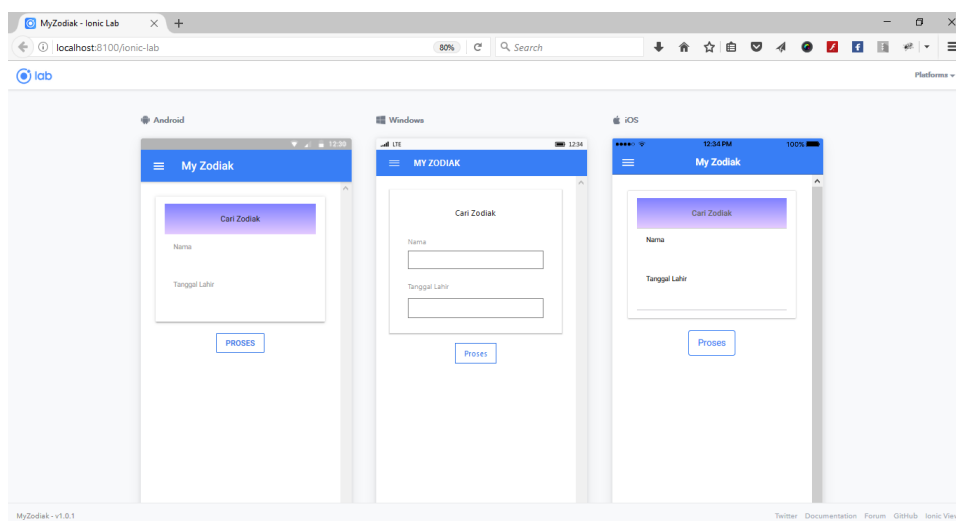
Penggunaan Ionic 2 Framework dan Angular 2 terbukti dapat dijalankan di mobile operating sistem yang berbeda.. Untuk menguji bahwa Ionic 2 telah *running* di *cross platform* dapat menggunakan *code* sebagai berikut:

- a. Pada Android: **ionic run android**
- b. Pada iOS : **ionic run ios**

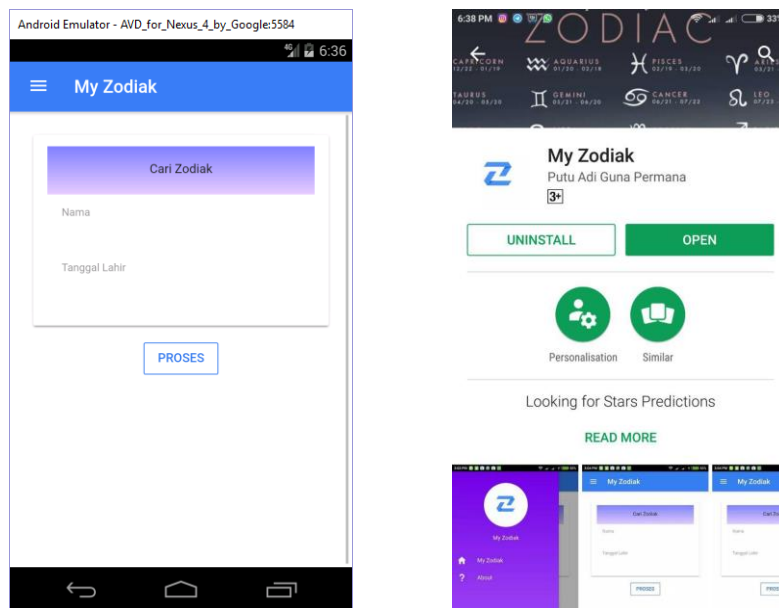
Dilihat dari sisi simulasi dalam bentuk web-based yang dimiliki Ionic, dapat menggunakan kode: `ionic serve --l`, maka akan tampil di browser tampilan beda platform.

### 3.2. Tampilan *User Interface* dari Aplikasi

Setelah merancang *user-interface* dan membuat panggilan API, pengembangan aplikasi tahap akhir siap untuk di-*deploy*. Pengembangan akhir aplikasi pada platform *mobile* dapat dilihat di bawah ini yang mengekstrak data *real time* dan menampilkannya ke pengguna untuk mengetahui situasi layanan yang sedang digunakan. Penulis menggunakan API Free dari `ibacor.com` untuk percobaan pembuatan aplikasi MyZodiak. Untuk hasil ke versi Android dapat di *download* di Google Play Store dengan kata kunci MyZodiak.



Gambar 2. Tampilan *User Interface* dari Aplikasi



Gambar 3. Tampilan pada *device* Android dan aplikasi pada Google Play Store

#### 4. Simpulan

Dengan bantuan Ionic 2 Framework, dapat dikembangkan aplikasi dengan fungsionalitas lengkap yang dapat dijalankan di *mobile operating system* yang berbeda seperti Android, iOS dan Windows Phone. Dapat disimpulkan bahwa Ionic sebagai teknologi baru, namun sangat berkembang, adalah sederhana dan mudah digunakan. Ionic 2 memungkinkan pengembang untuk membuat kode pada TypeScript, HTML dan CSS untuk menghasilkan aplikasi lintas platform *hybrid* yang secara penuh memiliki fungsional bagi pengguna.

Menciptakan teknologi yang efektif, dengan dimediasi melalui pendekatan *prototype (prototyping)* dimana pengembangan dan pengujian terhadap model kerja (prototipe) dari aplikasi yang dikembangkan telah melalui proses interaksi secara berulang dan melalui perbaikan terus-menerus hingga tercapai kesesuaian dengan yang dibutuhkan oleh pengguna (*user*). Aplikasi yang tepat tentunya sangat membantu pengguna dalam kelancaran maupun kecepatan penyelesaian berbagai tugas, baik dalam skala kecil atau besar.

Hasil penelitian dapat digunakan sebagai salah satu referensi di dalam pengembangan perangkat lunak, baik di Indonesia maupun dalam skala global, untuk melakukan efisiensi didalam proses pembuatan perangkat lunak berbasis *mobile*.

#### Daftar Pustaka

- [1] Aarush Gupta, Abdul Gaffar H, 2016, *Hybrid Application Development using Ionic Framework & AngularJS*, Proceeding of International Journal of Innovative Research in Computer Science & Technology, pp. 62 – 64, 2016.
- [2] Ari Lerner, 2016, *Ng-Book – the Complete Book on AngularJS*, Fullstack.io, 2013.
- [3] Arvind Ravulavaru, *Learning Ionic – Build Hybrid Mobile Applications with HTML5*, 2105
- [4] Carstensen, P.H., and Vogelsang, L., “Challenges in Web Development” in 9th European Conference of Information Systems, 2001.
- [5] Chris Griffith, 2017, *Mobile App Development with Ionic 2 Cross-Platform Apps with Ionic, Angular, and Cordova*, Australia, April 2017.
- [6] Ketan Anant More, Ms.Priya Chandran, 2016, *Native Vs Hybrid Apps*, Proceeding of International Journal of Current Trends in Engineering & Research, pp. 563 – 572, June 2016.
- [7] Ms.M.Gowthami, Rakshana.R, Alagammai.P, Sapna.A, 2017, *Creation of Password Management App Using Ionic Framework and Firebase*, Proceeding of International Journal of Advanced Research in Basic Engineering Sciences and Technology, pp. 318 – 328, February 2017.
- [8] Hoc Phan, *Full Stack Mobile App with Ionic Framework*, 2014.
- [9] Safaat H.Nazruddin. *Pemrograman Aplikasi Mobile Smartphone dan Tablet PC berbasis Android*. Informatika. Bandung. 2012.