

## APLIKASI HYBRID PADA SISTEM INFORMASI PENYEWAAN BUKU

**Dimas Iqbal Pradana**

Fakultas Sains dan Matematika, Departemen Ilmu Komputer / Informatika  
Universitas Diponegoro  
Email: dimasiqbal69@gmail.com

**Indra Waspada**

Fakultas Sains dan Matematika, Departemen Ilmu Komputer / Informatika  
Universitas Diponegoro  
Email: indrawaspada@undip.ac.id

### ABSTRAK

Saat ini persewaan buku konvensional mengalami permasalahan dalam manajemen data, pemberian informasi koleksi buku, serta transaksi penyewaan yang mengharuskan penyewa untuk mendatangi tempat persewaan buku. Untuk itu diperlukan suatu sistem informasi yang dapat mendukung transaksi penyewaan. Penggunaan *smartphone* yang memiliki akses internet sudah menjadi hal umum. Pemanfaatan aplikasi *mobile* yang mendukung akses internet dalam pengembangan sistem informasi penyewaan buku dapat memberi kemudahan penggunaan dan akses informasi yang cepat bagi penyewa buku. Beragamnya jenis *mobile platform* menyebabkan pengembangan aplikasi *mobile native* menjadi tidak efisien, baik dalam aspek waktu maupun biaya pengembangan. Sebagai solusinya, teknologi *mobile hybrid* dapat mengatasi masalah tersebut. Berdasarkan latar belakang tersebut maka dibangun aplikasi *mobile hybrid* pada sisi klien sistem informasi penyewaan buku menggunakan *framework* Ionic. Klien ini terhubung dengan aplikasi *back-end web administrator* melalui *RESTful Web Service*.

**Kata kunci:** aplikasi *hybrid*; sistem informasi penyewaan buku; *framework* ionic.

### ABSTRACT

*Traditional book rentals have problems in data management, providing information on book collections, and leasing transactions that require tenants to visit a book rental place. For this reason, an information system is needed that can support rental transactions. The use of smartphones that have internet access has become a common thing. The use of mobile applications that support internet access in the development of book rental information systems can provide ease of use and quick access to information for book tenants. The variety of mobile platforms causes the development of native mobile applications to be inefficient, both in terms of time and development costs. As a solution, mobile hybrid technology can overcome these problems. Based on this background, a hybrid mobile application was built on the client side of the book rental information system using the Ionic framework. This client is connected to a web administrator's back-end application through the RESTful Web Service.*

**Keywords:** *hybrid application; book rental information system; ionic framework.*

## 1. PENDAHULUAN

Minat baca pada masyarakat Indonesia dibandingkan negara lain masih sangat rendah. Masyarakat Indonesia mempunyai frekuensi membaca dengan rata-rata hanya tiga sampai empat kali per minggu. Sementara itu jumlah buku yang dibaca rata-rata per tahun hanya lima hingga sembilan buku [1]. Dalam meningkatkan minat baca, perpustakaan merupakan fasilitator yang berperan mempermudah pengguna dalam mencari informasi dengan memberikan layanan peminjaman dan pengembalian buku yang merupakan layanan intinya [2].

Setiap perpustakaan memberikan layanan yang terbatas untuk kalangan tertentu sesuai jenis perpustakaan. Selain perpustakaan, taman baca atau persewaan buku juga berfokus dalam meningkatkan minat baca dan mengembangkan budaya baca [3]. Taman baca atau persewaan buku bertujuan untuk memberi kemudahan akses kepada warga masyarakat untuk memperoleh bahan bacaan [4]. Dengan layanan persewaan buku yang dapat diperoleh semua kalangan masyarakat, taman baca dapat melengkapi keterbatasan kalangan yang dilayani perpustakaan.

Persewaan buku yang menggunakan sistem konvensional masih mengalami permasalahan antara lain manajemen data yang kurang efektif dan efisien [5], keterbatasan pemberian informasi tentang koleksi buku yang dimiliki [6], serta transaksi penyewaan dan pengembalian yang mengharuskan penyewa untuk mendatangi tempat persewaan buku [7]. Untuk itu sebuah sistem informasi diperlukan untuk mendukung aktivitas yang dilakukan oleh persewaan buku. Sistem informasi penyewaan buku dapat memberikan kemudahan manajemen data bagi pengelola, kemudahan pemberian informasi koleksi buku, serta melayani transaksi penyewaan dan pengembalian bagi penyewa.

Hasil riset terkait pengguna internet di Indonesia sampai akhir tahun 2014 terus mengalami peningkatan hingga mencapai angka 88,1 juta orang. Sedangkan akses internet di Indonesia melalui *smartphone* sebanyak 85 persen [8]. Hasil riset tersebut menunjukkan bahwa saat ini penggunaan *smartphone* yang dilengkapi akses internet sudah menjadi hal yang umum. Berdasarkan hal tersebut pemanfaatan aplikasi *mobile* yang mendukung akses internet dalam pengembangan sistem informasi penyewaan buku dapat memberi kemudahan penggunaan dan akses informasi yang cepat bagi penyewa buku.

Peningkatan jenis *mobile platform* (Android, Ios, Windows Phone) menyebabkan pengembangan aplikasi *mobile native* menjadi tidak efisien baik dalam hal waktu dan biaya pengembangan [9]. Berdasarkan penelitian, implementasi aplikasi *hybrid* lebih menjanjikan solusi pengembangan aplikasi *mobile*. Aplikasi *hybrid* menggabungkan kelebihan *web* dan aplikasi *native*. Aplikasi *hybrid* dikembangkan menggunakan HTML5 dan JavaScript tanpa perlu adanya pengetahuan terperinci pada target *platform*. Keuntungan utama dari aplikasi *hybrid* adalah kemampuannya untuk menjalankan *source code* dalam berbagai *platform* dan pengembangannya *source code* yang menggunakan teknologi pengembangan *web* yang banyak digunakan [9].

Oleh karena itu guna mengatasi penggunaan beragam *platform smartphone*, Aplikasi *hybrid* lebih tepat dalam pengembangan sistem informasi penyewaan buku pada perangkat *mobile*. Adanya aplikasi *hybrid* pada sistem informasi penyewaan buku diharapkan dapat memberikan kemudahan kepada penyewa dalam melakukan aktivitas penyewaan melalui *smartphone* tanpa adanya kendala keterbatasan *mobile platform* yang digunakan.

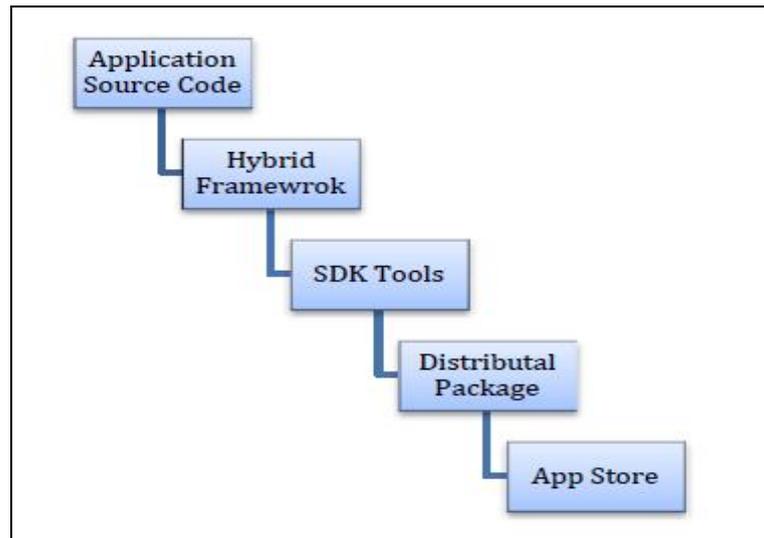
### **1.1 Pengertian Sewa**

Menurut Kamus Besar Bahasa Indonesia sewa adalah pemakaian sesuatu dengan membayar uang atau uang yang dibayarkan karena memakai atau meminjamkan sesuatu [10]. Sedangkan menurut R. Subekti dan R. Tjirosudibyo sewa-menyewa adalah suatu perjanjian dengan pihak yang mengikat dirinya untuk memberikan barang selama waktu tertentu dengan pembayaran suatu harga yang oleh pihak terakhir disanggupi pembayarannya [11].

### **1.2 Aplikasi Hybrid**

Aplikasi *hybrid* adalah aplikasi yang menggabungkan teknologi aplikasi *web* dan aplikasi *native*, tetapi tampilan dan cara kerja aplikasi *hybrid* lebih seperti aplikasi *web* daripada aplikasi *mobile*. Melalui pendekatan *hybrid*, pengembang mengkodekan *framework* nya sendiri, mengambil keuntungan dari alat pengembang yang sudah ada [12].

Aplikasi *hybrid* dibangun menggunakan HTML5, dan tidak memerlukan pengetahuan yang terperinci bagi target *platform*. Pendekatan *hybrid* mengambil keuntungan dari *browser* dan kemampuan perangkat *mobile*. Pendekatan *hybrid* dapat digunakan untuk aplikasi yang didukung dengan *server* maupun aplikasi *standalone*. Keuntungan utama dari pendekatan *hybrid* adalah penggunaan antarmuka *user* di *platform* yang berbeda memanfaatkan fitur *platform native*. Aplikasi dapat menggunakan fitur perangkat *native* yang tersedia menggunakan API [13]. Alur pengembangan aplikasi *hybrid* bisa dilihat pada Gambar 1.



**Gambar 1. Alur Pengembangan Aplikasi Hybrid** [12]

Berbagai aplikasi *mobile* memiliki perbedaan tersendiri dalam pengembangannya, baik dari bahasa pemrograman yang digunakan sampai fitur yang disediakan. Analisa perbandingan ketiga pendekatan tersebut dapat dilihat pada Tabel 1.

**Tabel 1. Perbandingan pendekatan pengembangan aplikasi mobile** [12]

<i>Fitur</i>	<i>Aplikasi Native</i>	<i>Aplikasi Hybrid</i>	<i>Aplikasi Web</i>
Bahasa Pemrograman	Hanya <i>Native</i>	<i>Native</i> dan <i>Web</i> atau hanya <i>Web</i>	Hanya <i>Web</i>
Akses Perangkat	Penuh	Penuh	Sebagian
Kecepatan	Sangat Cepat	Kecepat <i>Native</i>	Cepat
Biaya Pengembangan	Mahal	Masuk Akal	Masuk Akal
<i>App Store</i>	Ya	Ya	Tidak
Kecanggihan Grafik	Tinggi	Sedang	Sedang
Fleksibilitas <i>Upgrade</i>	Rendah	Sedang	Tinggi
Contoh Aplikasi	<i>Angry Birds</i>	<i>Ebay, Paypal</i>	<a href="http://m.facebook.com">http://m.facebook.com</a> , <a href="http://m.bbc.com">http://m.bbc.com</a> <i>m/</i>

### 1.3 Restful Web Service

Menurut Roy Thomas Fielding, REST (*Representational State Transfer*) merupakan arsitektur aplikasi perangkat lunak yang memodelkan cara data direpresentasikan, diakses, dan dimodifikasi pada *web*. Data dan fungsi pada arsitektur REST diakses dengan menggunakan *Uniform Resource Identifiers* (URIs) dan dianggap sebagai sumber daya. URIs biasanya merupakan *link* pada *web*. Arsitektur REST merupakan arsitektur *client-server*, dan dirancang untuk menggunakan protokol komunikasi *stateless*, biasanya berupa HTTP. Pada arsitektur REST, antarmuka dan protokol standar digunakan *client* dan *server* untuk menukarkan representasi sumber daya yang ada [14].

*RESTful web service* merupakan aplikasi *web* yang dibangun berdasarkan arsitektur REST. *RESTful* menampilkan sumber daya berupa data dan fungsi melalui URI dari *web*, dan menggunakan 4 metode utama HTTP untuk *create*, *retrieve*, *update* dan *delete* sumber daya (CRUD) [14]. Metode HTTP dan aksi CRUD bisa dilihat pada Tabel 2.

**Tabel 2. Metode HTTP dan aksi CRUD** [14]

<i>No.</i>	<i>Metode HTTP</i>	<i>Aksi CRUD</i>
1	<i>GET</i>	<i>Retrieve resource</i>
2	<i>POST</i>	<i>Create resource</i>
3	<i>PUT</i>	<i>Update resource</i>
4	<i>DELETE</i>	<i>Delete resource</i>

## 2. METODOLOGI PENELITIAN

### 2.1 Pengumpulan Data

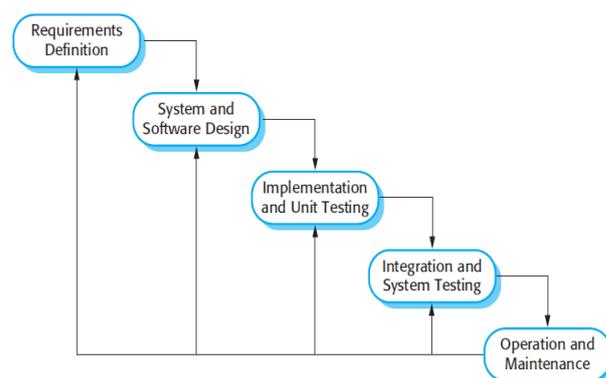
Dalam penelitian ini data yang diperlukan diperoleh menggunakan 2 macam cara sebagai berikut :

- a) Studi Pustaka  
Studi pustaka dilakukan dengan mempelajari berbagai referensi yang terkait dengan topik penelitian yaitu tentang aplikasi *hybrid* dan sistem informasi penyewaan buku. Referensi terkait penelitian didapatkan melalui berbagai macam media antara lain buku, jurnal ilmiah maupun laporan penelitian.
- b) Wawancara  
Wawancara dilakukan dalam bentuk tanya jawab secara langsung kepada orang-orang yang memiliki informasi yang diperlukan sesuai dengan penelitian ini. Wawancara dilakukan terhadap pemilik tempat persewaan buku Bulan Biru dan pemilik tempat persewaan buku Garfield.

### 2.2 Metode Pengembangan Sistem

Metode pengembangan sistem pada penelitian ini menggunakan model *Waterfall*. Model *Waterfall* dikenal karena urutan yang menurun dari satu fase ke fase yang lainnya. Bagan model *Waterfall* dapat dilihat pada Gambar 2. Tahap-tahap utama dari model *Waterfall* menjelaskan kegiatan-kegiatan pengembangan dasar [15]:

- a) *Requirements definition and analysis*. Pada tahap ini dilakukan kegiatan untuk menentukan layanan sistem, batasan, dan tujuan sistem melalui konsultasi dengan *user* sistem. Kebutuhan yang telah didapatkan kemudian dijelaskan secara rinci dan bertugas sebagai spesifikasi sistem.
- b) *System and software design*. Pada tahap ini proses perancangan perangkat lunak dilakukan untuk mengalokasikan kebutuhan sistem untuk menentukan keseluruhan arsitektur sistem. Perancangan perangkat lunak melingkupi identifikasi dan deskripsi mendasar tentang abstraksi sistem perangkat lunak dan berbagai hubungannya.
- c) *Implementation and unit testing*. Pada tahap ini perancangan perangkat lunak diwujudkan sebagai seperangkat program atau unit program. Pengujian unit dilakukan untuk memverifikasi setiap unit telah memenuhi spesifikasi yang ada.
- d) *Integration and system testing*. Pada tahap ini unit program atau program individual diintegrasikan dan diuji sebagai sistem yang lengkap untuk menjamin bahwa kebutuhan sistem telah terpenuhi. Setelah pengujian selesai, perangkat lunak dikirim kepada pelanggan.
- e) *Operation and maintenance*. Pada tahap ini biasanya merupakan fase siklus hidup yang paling lama. Sistem di-*install* dan dipakai langsung. Pemeliharaan sistem melingkupi perbaikan *error* sistem yang tidak ditemukan pada tahap-tahap sebelumnya, peningkatan implementasi unit sistem dan pelayanan sistem, sementara kebutuhan baru ditambahkan.



Gambar 2 Model *Waterfall* [15]

### 2.3 Object Oriented Analysis and Design

*Object Oriented* adalah cara untuk mengembangkan perangkat lunak dengan membangun modul atau objek yang dapat dengan mudah digantikan, diubah atau digunakan kembali. *Object oriented* memandang sistem sebagai objek yang saling berkooperatif dan berkolaborasi. Pengembangan *object oriented* memandang perangkat lunak sebagai koleksi objek diskrit yang merangkum data serta fungsinya untuk memodel objek dunia nyata [16].

Proses analisis pada *object oriented* dilakukan menggunakan *requirement* yang dikenal secara beragam sebagai *requirement specification*. Dengan menggunakan *requirement specification*, sistem analis menciptakan model sistem dan memungkinkan identifikasi beberapa komponen sistem dan hubungan di antara mereka. Produk akhir fase ini adalah model konseptual kelas sistem yang menggambarkan fungsionalitas sistem, mengidentifikasi entitas dan mencatat sifat hubungan antara entitas-entitas ini [17].

Pada proses desain dilakukan dengan menggunakan struktur kelas yang dihasilkan oleh analisis untuk merancang perilaku sistem sesuai model. Alat UML yang paling utama digunakan adalah *sequence diagram* untuk menentukan detail tentang bagaimana perilaku model akan direalisasikan. Kemudian digunakan *class diagram* untuk memodelkan hubungan antar kelas, dilanjutkan dengan desain antarmuka dan *database* [17].

### 3. HASIL DAN PEMBAHASAN

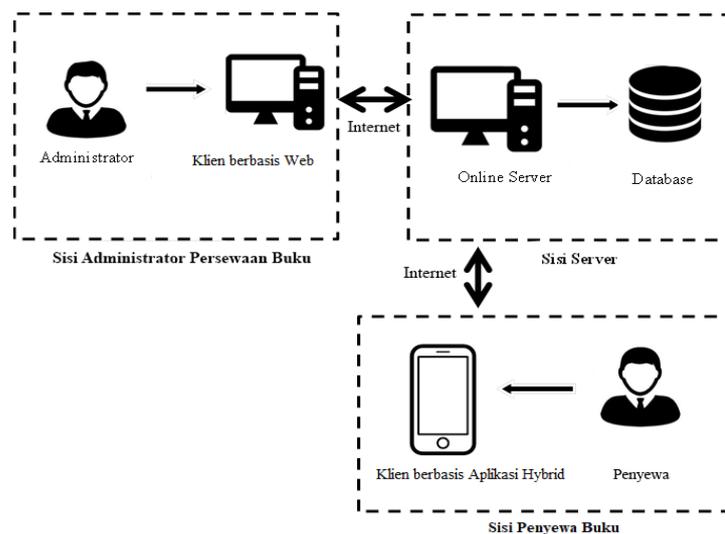
#### 3.1 *Bisnis Model Penyewaan Buku Konvensional*

Pada tahap pengumpulan *requirement* sistem informasi penyewaan buku, dilakukan wawancara langsung kepada narasumber penelitian ini yaitu persewaan buku Bulan Biru dan persewaan buku Garfield. Dari hasil wawancara dapat dihasilkan kesimpulan sebagai berikut :

- Kedua tempat persewaan buku hanya menyediakan informasi buku secara langsung jika penyewa mengunjungi tempat persewaan buku.
- Proses penyewaan buku di tempat persewaan buku dilakukan dengan mendatangi tempat persewaan secara langsung, memilih buku dan membayar biaya sewa.
- Proses pengembalian buku di kedua tempat persewaan buku dilakukan dengan mendatangi tempat persewaan secara langsung.
- Keterlambatan buku dibayarkan saat penyewa datang untuk mengembalikan buku.
- Kedua tempat persewaan buku pada saat dilakukan wawancara belum mempunyai sistem berbasis komputer untuk menyewakan buku.

#### 3.2 *Arsitektur Sistem*

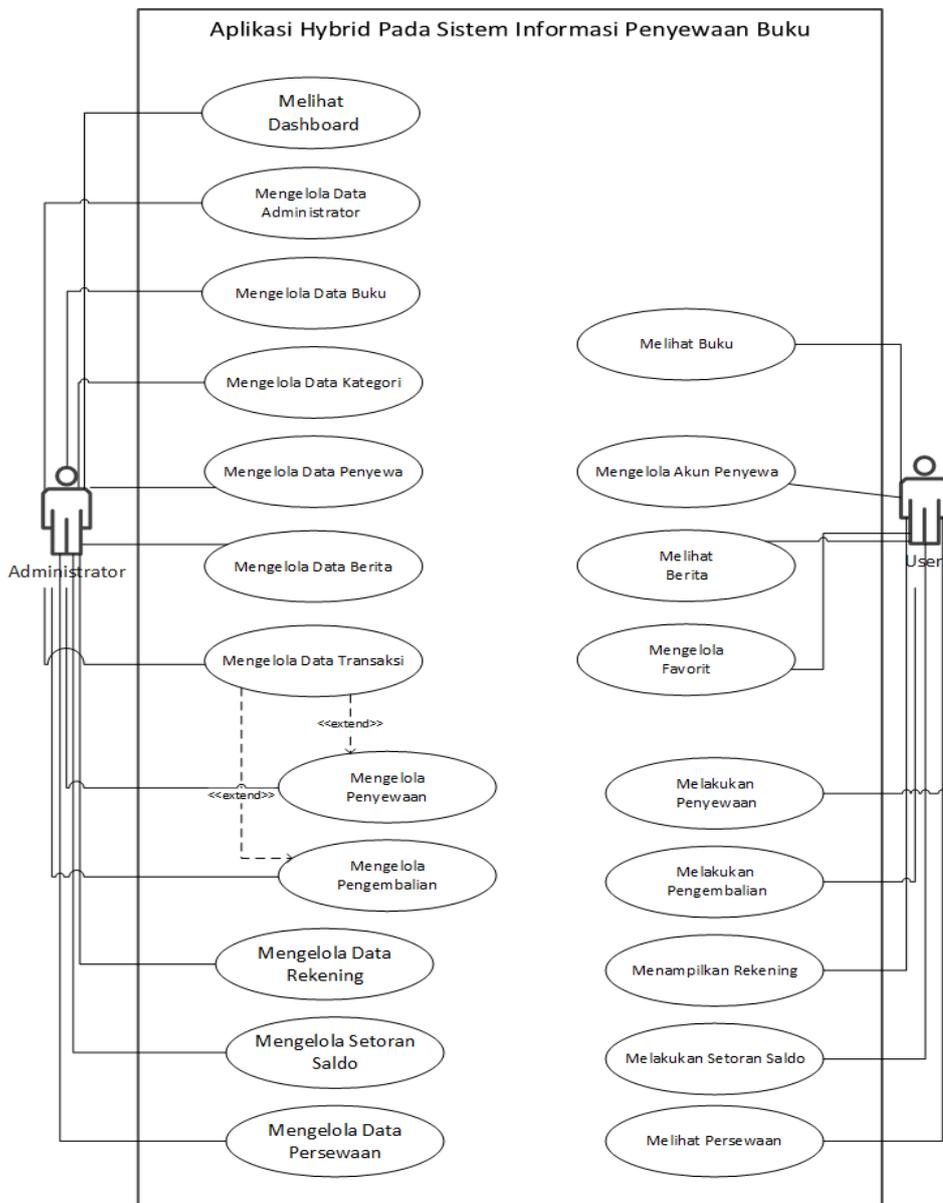
Aplikasi *hybrid* pada sistem informasi penyewaan buku merupakan sistem yang mempunyai dua pengguna. Pengguna tersebut meliputi *administrator* dan penyewa. Sistem informasi pada sisi *administrator* merupakan sistem yang berbasis *web* sedangkan pada sisi penyewa menggunakan aplikasi *hybrid*. Setiap pengguna memiliki peran yang berbeda satu sama lain. Untuk dapat mengakses sistem tersebut setiap pengguna membutuhkan akses ke jaringan internet, seperti terlihat pada arsitektur sistem pada Gambar 3.



Gambar 3. Arsitektur Aplikasi *Hybrid* Pada Sistem Informasi Penyewaan Buku

### 3.3 Use Case Diagram

Analisis *use case* dilakukan terhadap pengguna (*user*) serta fungsi-fungsi utama aplikasi *hybrid* pada sistem informasi penyewaan buku. Dari analisis yang dilakukan, permasalahan yang ada ditransformasikan ke dalam *detail use case* sistem yang akan dibangun.

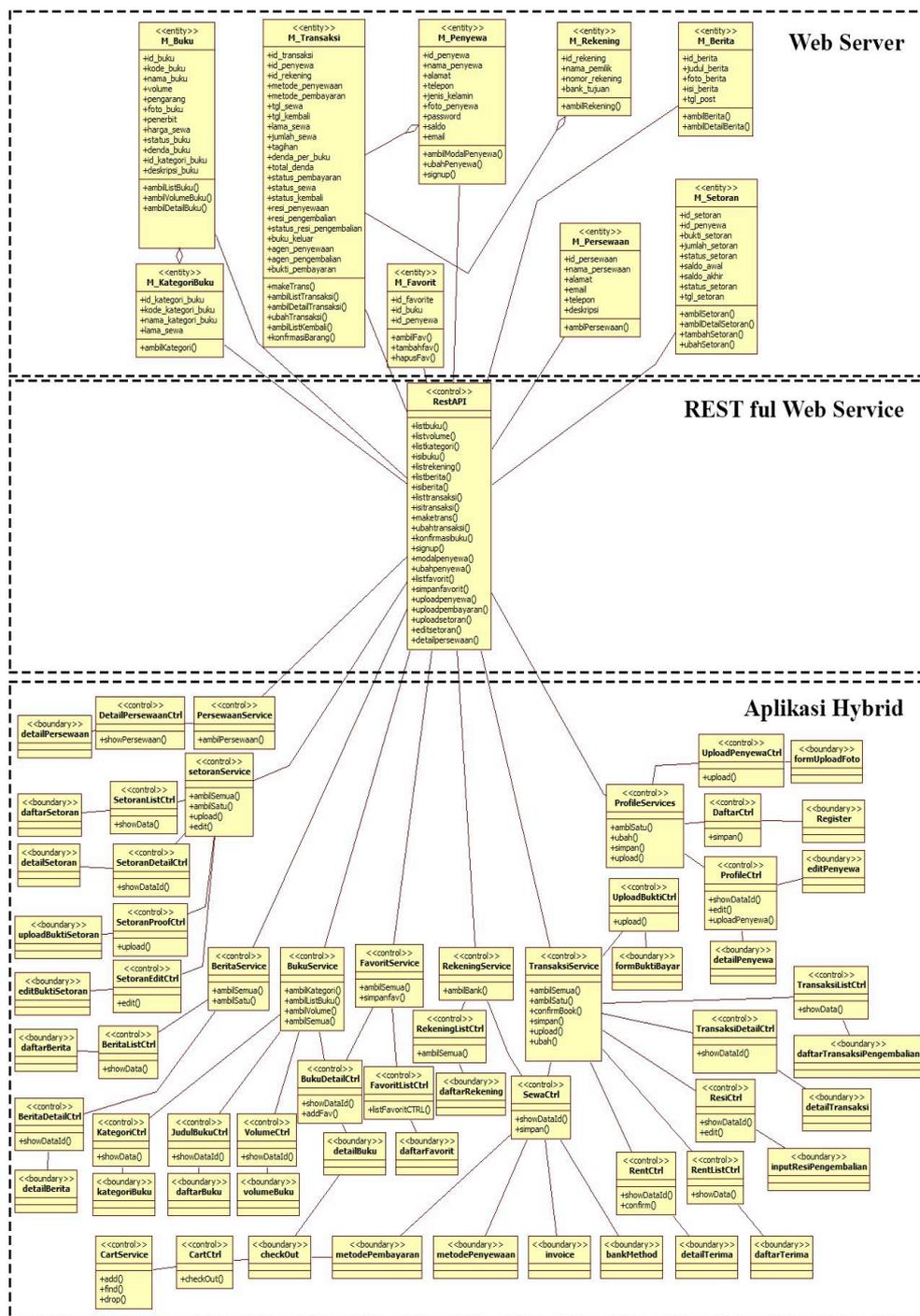


Gambar 4. Use Case Diagram Aplikasi Hybrid Pada Sistem Informasi Penyewaan Buku

### 3.4 Class Diagram

Setiap data kelas dapat dimodelkan hubungannya dengan *class diagram*. Pada *class diagram* ini menjelaskan hubungan antar kelas yang telah dijelaskan pada *sequence diagram*. *Class diagram* pada *web administrator* terdiri dari 9 *class entity* yang berfungsi untuk mengolah data ke dalam *database*. Masing-masing *class entity* terhubung dengan *class control* yang memproses data dari 34 *boundary* sesuai use case masing-masing. Sedangkan pada aplikasi *hybrid*, *entity* terhubung melalui *Restful Web Service* yang berupa *control RestAPI* yang berada pada *web administrator* beserta *control* dan *boundary* lainnya. *Class control* RestAPI berfungsi untuk memproses data yang dikirimkan maupun diminta oleh *class control* aplikasi *hybrid*. Perancangan *class diagram* pada *web administrator* ditunjukkan pada Gambar 5. Sedangkan *class diagram* keseluruhan ditunjukkan pada Gambar 6.





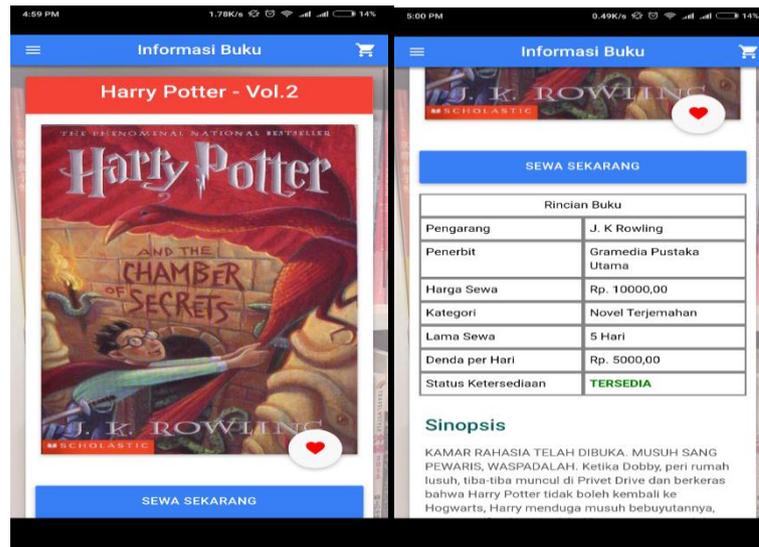
Gambar 6. Class Diagram Keseluruhan Sistem

### 3.5 Implementasi Sistem

Pada bagian ini disajikan hasil implementasi sistem yang meliputi antarmuka dari aplikasi *hybrid* dan *web administrator*.

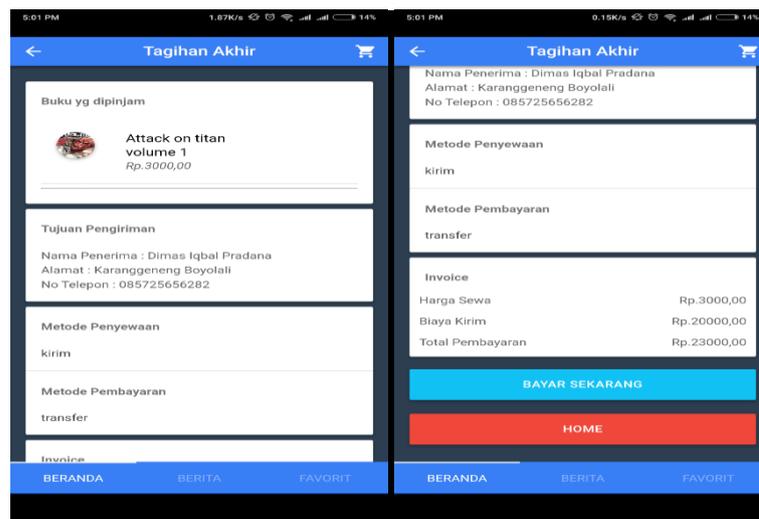
#### 3.5.1 Aplikasi Hybrid

Pada antarmuka aplikasi *hybrid class boundary* InfoBuku ditampilkan ketika aktor penyewa sudah melakukan *login* sistem. Agar dapat melihat buku, aktor penyewa harus memilih salah satu kategori buku, kemudian memilih judul buku, lalu *volume* buku. Adapun hasilnya berupa informasi buku ditampilkan pada Gambar 7.



Gambar 7. Antarmuka *Class Boundary* InfoBuku

Pada antarmuka aplikasi *hybrid class boundary Invoice* ditampilkan ketika aktor penyewa akan melakukan penyewaan. Agar dapat menyewa buku, actor penyewa harus memasukkan buku ke keranjang dan memilih metode penyewaan dan memilih metode pembayaran dan menyelesaikan penyewaan dengan menekan tombol bayar sekarang pada *Class Boundary Invoice*. Hasil dari input penyewaan ditampilkan pada Gambar 8.



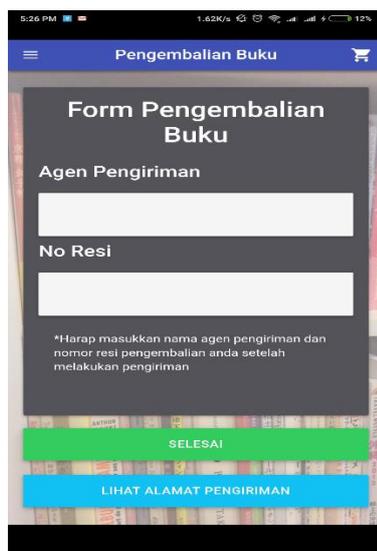
Gambar 8. Antarmuka *Class Boundary Invoice*

Pada antarmuka aplikasi *hybrid class boundary* detailTransaksi ditampilkan ketika aktor penyewa akan melihat transaksi pengembalian. Agar dapat melihat transaksi pengembalian, actor penyewa harus memilih menu daftar pengembalian dan memilih salah satu transaksi. Informasi pengembalian akan ditampilkan pada *class boundary* detailTransaksi. Hasil dari informasi transaksi pengembalian ditampilkan pada Gambar 9.



Gambar 9. Antarmuka *Class Boundary detail*Transaksi

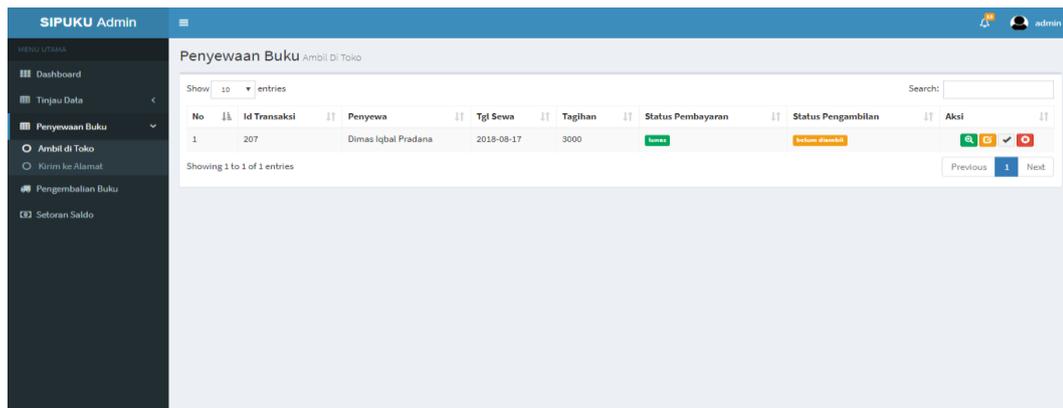
Pada antarmuka aplikasi *hybrid class boundary* inputResiPengembalian ditampilkan ketika aktor penyewa akan melakukan pengembalian. Agar dapat melakukan pengembalian, aktor penyewa harus mengisi form pengembalian buku dan menekan tombol selesai. Hasil dari informasi transaksi pengembalian ditampilkan pada Gambar 10.



Gambar 10. Antarmuka *Class Boundary* inputResiPengembalian

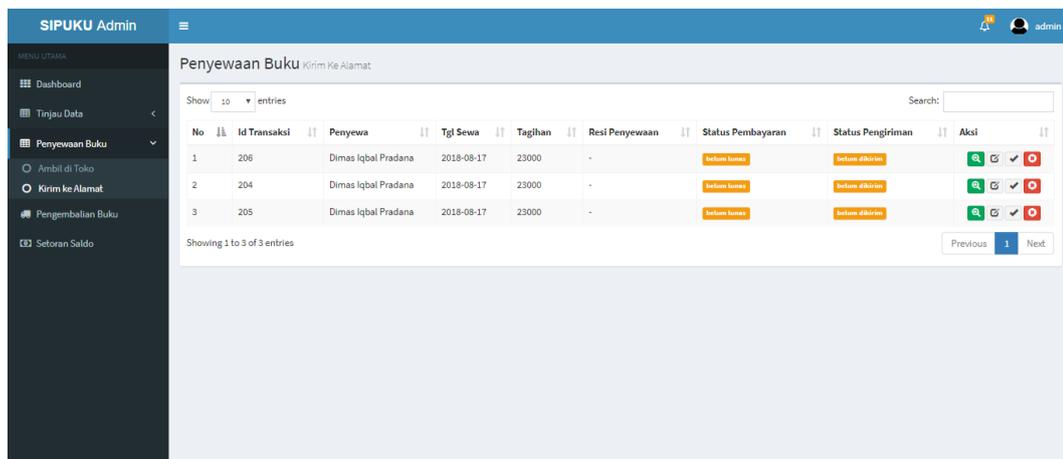
### 3.5.2 Web Administrator

Pada antarmuka *web administrator class boundary* hlmnPenyewaanAmbil ditampilkan ketika aktor *administrator* akan mengelola transaksi penyewaan berdasarkan metode penyewaan pengambilan di toko. Pada halaman ini aksi yang dapat dilakukan adalah melihat detail transaksi penyewaan ambil, mengubah status pembayaran ambil, mengubah status pengambilan, dan membatalkan transaksi penyewaan ambil yang dihasilkan aplikasi *hybrid*. Antarmuka *class boundary* hlmnPenyewaanAmbil ditampilkan pada Gambar 11.



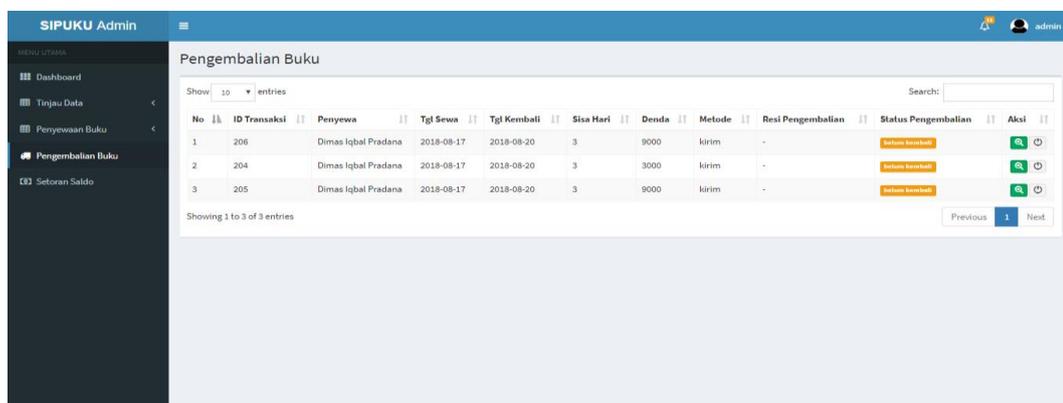
Gambar 11. Antarmuka *Class Boundary* hlmnPenyewaanAmbil

Pada antarmuka *web administrator class boundary* hlmnPenyewaanKirim ditampilkan ketika aktor *administrator* akan mengelola transaksi penyewaan berdasarkan metode penyewaan kirim ke alamat. Pada halaman ini aksi yang dapat dilakukan adalah melihat *detail* transaksi penyewaan kirim, mengubah status pembayaran kirim, mengubah status pengiriman, dan membatalkan transaksi penyewaan kirim yang dihasilkan aplikasi *hybrid*. Antarmuka *class boundary* hlmnPenyewaanKirim ditampilkan pada Gambar 12.



Gambar 12. Antarmuka *Class Boundary* hlmnPenyewaanKirim

Pada antarmuka *web administrator class boundary* hlmnPengembalian ditampilkan ketika aktor *administrator* akan mengelola transaksi pengembalian. Pada halaman ini aksi yang dapat dilakukan adalah melihat detail transaksi pengembalian yang dihasilkan aplikasi *hybrid* dan meng-*edit* status pengembalian. Antarmuka *class boundary* hlmnPengembalian ditampilkan pada Gambar 13.



Gambar 13. Antarmuka *Class Boundary* hlmnPengembalian

### 3.6 Pengujian

Pelaksanaan pengujian berdasarkan *platform* dilakukan pada simulator yang telah disediakan oleh *tool* Intel XDK. Pengujian dilakukan pada 3 *platform* yang berbeda. Berikut spesifikasi simulator yang digunakan dalam melakukan pengujian.

- a) Apple Iphone 6
  - Device* : Apple Iphone 6
  - Manufacturer* : Apple
  - OS* : iOS
  - OS Version* : 8
  - Screen* : 750x1334
  - Density* : 326 ppi
  - CSS Pixel Ratio* : 2
  - CSS Pixels* : 375x667
- b) Nokia Lumia 920
  - Device* : Nokia Lumia 920
  - Manufacturer* : Nokia
  - OS* : Windows Phone
  - OS Version* : 8
  - Screen* : 768x1280
  - Density* : 332 ppi
  - CSS Pixel Ratio* : 2
  - CSS Pixels* : 384x640
- c) Lenovo K900
  - Device* : Lenovo K900
  - Manufacturer* : Lenovo
  - OS* : Android
  - OS Version* : 4.2
  - Screen* : 1080x1920
  - Density* : 401 ppi
  - CSS Pixel Ratio* : 2.5
  - CSS Pixels* : 432x768

Selanjutnya dari pengujian ketiga *platform* tersebut maka dihasilkan hasil uji *use case* yang dijabarkan pada tabel berikut.

**Tabel 3. Pengujian *platform* berdasarkan *use case***

<i>Use Case</i>	<i>Android 4.2</i>	<i>IOS 8</i>	<i>Windows Phone 8</i>
Melihat buku	✓	✓	✓
Mengelola akun penyewa	✓	✓	✓
Melihat berita	✓	✓	✓
Melakukan penyewaan	✓	✓	✓
Melakukan pengembalian	✓	✓	✓
Mengelola favorit	✓	✓	✓
Melihat rekening	✓	✓	✓
Melakukan setoran saldo	✓	✓	✓
Melihat persewaan	✓	✓	✓

Keterangan:

- ✓ : Dapat Bekerja
- X : Tidak Bekerja

Kemudian pada pengujian *platform*, dilakukan pengujian terhadap akses perangkat oleh aplikasi dengan menggunakan *plugin*. Berikut penjabaran pengujian terhadap akses perangkat aplikasi.

**Tabel 4. Pengujian platform berdasarkan plugin**

Akses Perangkat	Plugin	Android 4.2	IOS 8	Windows Phone 8
Kamera	<i>cordova-plugin-camera</i>	✓	✓	✓
File	<i>cordova-plugin-file</i>	✓	✓	✓

Keterangan:

✓ : Dapat Bekerja

X : Tidak Bekerja

### 3.7 Analisis Hasil Pengujian

Pada sistem yang dibangun dilakukan dua jenis pengujian yaitu pengujian fungsional secara *blackbox* dan pengujian berdasarkan *platform*. Berdasarkan hasil pengujian fungsionalitas sistem, secara umum fungsionalitas aplikasi *hybrid* pada sistem informasi penyewaan buku telah memenuhi *use case* yang telah diidentifikasi sebelumnya. Sedangkan pada pengujian *platform* berdasarkan *use case* dan *plugin*, aplikasi *hybrid* telah berhasil diujikan di beberapa *platform* seperti Android, iOS, dan Windows Phone dengan hasil uji semua *use case* dan *plugin* berhasil dijalankan. Aplikasi *hybrid* juga membutuhkan akses internet untuk mengirim maupun menerima data dari *web administrator* menggunakan RESTful *web service*. *Web administrator* digunakan oleh *administrator* dalam mengelola data terkait penyewaan buku yang digunakan oleh aplikasi *hybrid*.

## 4. KESIMPULAN

Dari uraian yang telah dibahas pada bab-bab sebelumnya, maka dapat diambil kesimpulan sebagai berikut :

- Dihasilkan aplikasi *hybrid* pada sistem informasi penyewaan buku yang terdiri *web administrator* yang menangani pengolahan data terkait penyewaan, serta aplikasi *hybrid* yang digunakan untuk mengakses informasi terkait penyewaan buku dan melakukan penyewaan melalui *mobile smartphone*. Kedua sisi program dihubungkan dengan menggunakan *Restful Web Service*.
- Dalam perancangan sistem ini digunakan model proses *waterfall* melalui 4 tahapan yaitu fase *requirement definition* yang mengidentifikasi kebutuhan awal dari tinjauan pustaka dan wawancara yang dilakukan pada 2 tempat persewaan buku. Fase *system and software design* yang fokus dalam menentukan arsitektur sistem dan permodelan sistem *web administrator* dan aplikasi *hybrid*. Fase *implementation and unit testing* yang dilakukan untuk mengimplementasi desain pada perangkat lunak dimana *web administrator* dikembangkan menggunakan *framework Codeigniter* dan aplikasi *hybrid* dikembangkan menggunakan *framework ionic*. Fase *integration and system testing* dimana dilakukan pengujian fungsionalitas sistem menggunakan *black box testing*.
- Pengujian aplikasi *hybrid* pada ketiga *platform* telah berhasil dilakukan dengan menggunakan simulator Intel XDK dengan hasil semua *use case* dapat dijalankan dan semua *plugin* dapat bekerja.

## DAFTAR PUSTAKA

- Pratiwi, P.S. (2018). *Minat Baca Masyarakat Indonesia Masih Rendah*. Jakarta: CNNIndonesia.
- Astuti, P.D. 2015. "Peran Perpustakaan Dan Arsip Dalam Meningkatkan Minat Membaca Masyarakat Di Perpustakaan Umum Kota Bontang". *eJournal Ilmu Pemerintahan* 3, 1240–53.
- Haklev, S. (2008). *Mencerdaskan Bangsa – Suatu Pertanyaan Fenomena Taman Bacaan Di Indonesia*. Scarborough: University of Toronto.
- Suwanto, S.A. 2017. "Pengelolaan TBM Sebagai Sarana Meningkatkan Minat Baca Masyarakat". *ANUVA Undip* 1, 19–32.
- Agung Nugroho, B. (2015). *Perancangan Sistem Aplikasi Rental Pada Taman Bacaan Winda*. Semarang: Universitas Dian Nuswantoro.
- Purnamasari, I. 2012. "Perancangan Sistem Informasi Peminjaman Buku Dan Komik Pada Taman Bacaan Fortune Baleharjo Pacitan". *Sentra Penelitian Engineering dan Edukasi* 4, 10–4.
- Lestari, A.P. (2015). *Sistem Informasi Rental Buku (Studi Kasus: Afro Rental Buku)*. Yogyakarta: Universitas Gajah Mada.
- APJII. (2014). *Profil Pengguna Internet Indonesia 2014*. Jakarta: Asosiasi Penyelenggara Jasa

Internet Indonesia.

- [9] Xanthopoulos, S., and Xinogalos, S. (2013). *A Comparative Analysis Of Cross-Platform Development Approaches For Mobile Applications*. Thessaloniki: 6th Balkan Conference in Informatics.
- [10] Tim Penyusun Kamus Pusat Bahasa. (2008). *Kamus Bahasa Indonesia*. Jakarta: Pusat Bahasa.
- [11] Subekti, R., and Tjitrosudibyo, R. (2009). *Kitab Undang Undang Hukum Perdata (Burgerlijk Wetboek)*. Jakarta: PT. Pradnya Paramitra.
- [12] Tun, P.M. 2014. "Choosing A Mobile Application Development Approach". *Asean Journal of Management & Innovat* 1, 69–74.
- [13] S.Thakare, B. et al. 2014. "State Of Art Approaches To Build Cross Platform Mobile Application". *International Journal of Computer Applications* 107, 22–3.
- [14] Hamad, H. et al. 2010. "Performance Evaluation Of Restful Web Services For Mobile Devices". *Int. Arab J. e-Technol.* 1, 72–8.
- [15] Sommerville, I. (2011). *Software Engineering, 9th Edition*. Massachusetts: Pearson Education.
- [16] Bahrani, A. (1992). *Object-Oriented System Development*. London: McGraw Hill International Edition.
- [17] Dathan, B., and Ramnath, S. (2011). *Object-Oriented Analysis And Design*. London: Springer.