

APLIKASI UJIAN ONLINE PADA MOBILE DEVICE ANDROID

Alvin Assianto Leiman¹, Andreas Handoyo², Agustinus Noertjahyana³
Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Kristen Petra
Jl. Siwalankerto 121 – 131 Surabaya 60336
Telp. (031) – 2983455, Fax. (031) – 8417658
Email: alvinleiman3@gmail.com¹, handoyo@petra.ac.id², agust@petra.ac.id³

ABSTRAK : Pada saat ini perkembangan handphone yang mendukung koneksi Internet dan teknologi Android semakin berkembang pesat. Aplikasi Android telah banyak digunakan pada handphone sebagai penunjang aktifitas sehari-hari. Salah satu contoh aplikasi yang bisa dikembangkan lebih lanjut ialah ujian online yang dapat digunakan sebagai media alternatif dalam pengerjaan ujian.

Pada skripsi ini dikembangkan aplikasi ujian online yang memungkinkan pengguna handphone untuk mengerjakan tes yang terdiri dari soal multiple choice dengan single maupun multi answer, dan soal essay, serta melihat hasil tes yang pernah diikutinya. Aplikasi ujian online menggunakan koneksi Internet untuk berkomunikasi dengan server. Aplikasi ujian online dibuat dengan menggunakan bahasa pemrograman Java.

Berdasarkan hasil pengujian, aplikasi ini dapat berjalan dengan baik pada *smartphone* dengan sistem operasi Android versi 4.0 (Ice Cream Sandwich) ke atas dalam orientasi *landscape*.

Kata kunci: *Web service*, SOAP, Ujian Online.

ABSTRACT : *Nowadays, cellular phone that supported by internet connection and Android technology is widely available. Many Android applications have been utilized used on cellular phones for various daily activities. Test Online is one example application which can be developed as an alternative for taking test.*

In this research, an online test application has been developed which enable users to take a particular test that consists of multiple choice items with single or multi answers and essay items, and also see their test result via their own cellular phones. This application will utilize internet connection to communicate with the server. This application was developed by using Java programming language.

Based on the testing result, this application is working properly on Smartphone with Android Operating System version 4.0 (Ice Cream Sandwich) or higher in landscape orientation.

Keywords: *Web service, SOAP, Online Test*

1. PENDAHULUAN

Smartphone merupakan telepon genggam yang mempunyai kemampuan tingkat tinggi bahkan terkadang memiliki fungsi yang menyerupai komputer. Saat ini *Smartphone* mengadopsi *Operating System* (OS) seperti Android, Blackberry OS, Windows Phone OS, dan masih banyak lagi. Kemampuan *Smartphone* sangat berkembang dengan pesat serta didukung dengan bentuknya yang kecil dan juga memiliki berat yang cukup ringan sehingga mudah dibawa kemana saja. *Smartphone* juga memiliki

kemampuan untuk mengakses aplikasi dengan menggunakan fasilitas internet dengan program yang relatif berukuran kecil. *Smartphone* yang banyak beredar saat ini menggunakan OS Android.

Android merupakan sebuah *Operating System*(OS) yang berbasis *Open Source* yaitu sebuah piranti lunak yang memakai basis kode komputer yang bisa didistribusikan secara terbuka sehingga pengguna bisa membuat aplikasi baru didalamnya. Android telah menguasai sebagian besar dari penggunaan OS di Indonesia menurut data yang dilansir *International Data Corporation* (IDC) pada tahun 2009. Alasan lain yang menyebabkan Android sangat diminati karena harga terjangkau, memiliki kemampuan mengakses internet, serta aplikasi yang dibuat lebih mudah untuk diaplikasikan pada *device*. Aplikasi yang ada saat ini kebanyakan memiliki kemampuan mengakses internet sehingga akses real-time ke data yang dibutuhkan. Saat ini banyak orang yang terhalang oleh ruang dan waktu, ujianpun masih terhalang dengan tempat dan juga waktu yang membuat tidak leluasa melakukan ujian. Ketidakhadiran dalam sebuah ujian dapat mengakibatkan ketidakhadiran mata pelajaran maupun mata kuliah tertentu. Oleh karena itu, aplikasi ujian ini ditujukan untuk memberi alternatif bagi siswa ujian yang kesulitan mengikuti ujian.

2. LANDASAN TEORI

2.1 Android

Android adalah sebuah sistem operasi yang dikembangkan untuk perangkat *mobile* dengan berbasis pada *Linux*. Awalnya, sistem operasi android dikembangkan oleh Android Inc. Tetapi pada tahun 2005, sistem operasi android dibeli oleh *Google*. Dalam pengembangan android di tahun 2007, dibentuk sebuah konsorsium bernama *Open Handset Alliance* (OHA) yang terdiri dari beberapa perusahaan yaitu *Texas Instruments, Broadcom Corporation, Google, HTC, Intel, LG, Marvel Technology Group, Motorola, Nvidia, Qualcomm, Samsung Electronics Sprint Nextel* dan *T-mobile* yang bertujuan untuk mengembangkan standar terbuka untuk perangkat *mobile*. Setahun kemudian, beberapa perusahaan lain ikut bergabung yaitu *PacketVideo, ARM Holdings, Atheros Communications, Asustek Computer Inc, Garmin Ltd, Sofibank, Sony Ericsson, Toshiba Corp, dan Vodafone Group Plc*. Android versi 1.1 yaitu android pertama, diluncurkan oleh *Google* pada tanggal 9 Maret tahun 2009 [1].

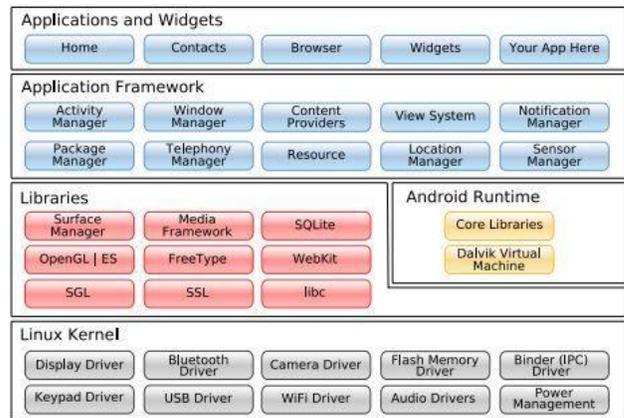
Fitur – fitur yang tersedia pada Android adalah sebagai berikut.

- *Framework* Aplikasi. *Framework* Aplikasi memungkinkan penggunaan dan pemindahan dari komponen yang tersedia.
- *Dalvik virtual machine*. *Dalvik virtual machine* merupakan *virtual machine* yang dioptimalkan untuk perangkat *mobile*.

- Grafik. Grafik yang didukung oleh android berupa grafik 2D dan grafik 3D yang didasarkan pada *library* OpenGL.
- SQLite. SQLite digunakan untuk penyimpanan data pada android.
- Media yang didukung android berupa audio, video, dan berbagai format gambar (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF).
- Android mendukung koneksi GSM, *Bluetooth*, EDGE, 3G dan WiFi tergantung dari spesifikasi *hardware* yang digunakan. Android mendukung fitur *Camera*, *Global Positioning System* (GPS), *compass* dan *accelerometer* tergantung dari spesifikasi *hardware* yang digunakan.
- Android memiliki lingkungan pengembangan yang kaya, termasuk *emulator*, peralatan *debugging*, dan *plug-in* untuk Eclipse IDE.

Sistem operasi android memiliki arsitektur yang terdiri dari empat lapisan seperti yang terlihat pada Gambar 1.

1. *Applications*. Lapisan ini adalah lapisan aplikasi, serangkaian aplikasi akan terdapat pada perangkat *mobile* termasuk aplikasi inti seperti kalender, kontak, SMS, dan lain sebagainya. Aplikasi-aplikasi ini ditulis dengan bahasa pemrograman Java.
2. *Application Framework*. Pengembang aplikasi memiliki akses penuh ke Android sama dengan aplikasi inti yang telah tersedia. Pengembang dapat dengan mudah mengakses informasi lokasi, mengatur alarm, menambahkan pemberitahuan ke *status bar* dan lainnya sebagainya. Arsitektur aplikasi ini dirancang untuk menyederhanakan penggunaan kembali komponen, aplikasi apa pun dapat mempublikasikan kemampuannya dan aplikasi lain dapat menggunakan kemampuan mereka sesuai batasan keamanan yang telah ditentukan. Dasar dari aplikasi adalah layanan dan sistem, yaitu berbagai *view* yang digunakan untuk membangun *user interface*, *content provider* yang memungkinkan aplikasi untuk berbagi data, *resource manager* yang menyediakan akses non-kode seperti grafik, string dan layout, *notification manager* yang membuat aplikasi dapat menampilkan tanda pada *status bar* dan *activity manager* yang berguna mengatur daur hidup dari aplikasi.
3. *Libraries*. *Libraries* ditulis dalam bahasa C/C++ dan akan digunakan oleh berbagai komponen pada sistem Android.
4. *Android Runtime*. *Android Runtime* merupakan *libraries* inti yang menyediakan sebagian besar fungsi yang tersedia di *libraries* inti dari bahasa pemrograman Java. Setiap aplikasi akan berjalan sebagai proses sendiri pada *Dalvik Virtual Machine (VM)*.
5. *Linux Kernel*. Android bergantung pada Linux versi 2.6 untuk layanan sistem inti seperti keamanan, manajemen memori, manajemen proses, *network stack* dan model *driver*. Kernel juga bertindak sebagai lapisan antara *hardware* dan seluruh *software*.



Gambar 1. Arsitektur Android

Sumber : Mudah Membuat Aplikasi Android (2011)

2.2 Simple Object Access Protocol

Simple Object Access Protocol (SOAP) adalah protokol untuk bertukar pesan dalam bentuk dokumen XML melalui jaringan komputer atau sebuah jalan untuk program yang berjalan pada suatu sistem operasi untuk berkomunikasi dengan program pada sistem operasi yang sama maupun berbeda dengan menggunakan HTTP dan XML sebagai mekanisme untuk pertukaran data. SOAP mendefinisikan protokol untuk jalur pesan XML searah atau *asynchronous* beserta gaya interaksi permintaan atau responsnya. SOAP berinteraksi melalui SOAP *nodes* yang dapat berupa pengirim pesan SOAP, penerima atau keduanya. Dalam sebuah kasus khusus, SOAP dapat berperan sebagai *intermediary* anatar pengirim dan penerima dengan tujuan untuk menangani *header special* [2].

2.3 Web services Description Language

Web services Description Language (WSDL) adalah sebuah dokumen XML yang dibuat untuk mendeskripsikan dan memuliskan format dan protokol dari sebuah *web services*. Elemen WSDL berisi sebuah deskripsi dari data, pada umumnya menggunakan satu atau lebih skema XML untuk diberikan ke *web services* sehingga baik pengirim maupun penerima mengenali data yang ditukarkan. Elemen WSDL juga berisi sebuah deskripsi dari operasi-operasi yang akan dijalankan pada data tersebut sehingga penerima pesan tahu bagaimana memprosesnya dan pengirim tahu harus bagaimana mengirimnya. Dalam berkomunikasi, pengirim dan penerima harus memiliki akses ke WSDL yang sama atau dalam satu skema XML agar dapat memahami satu sama lain [3].

2.4 JQuery

Menurut Lukmanul Hakim, JQuery adalah *javascript library* yaitu kumpulan kode atau fungsi javascript siap pakai, sehingga mempermudah dan mempercepat dalam membuat kode javascript [4]. Secara standar, apabila dalam membuat kode javascript, maka diperlukan kode yang panjang, bahkan terkadang sulit untuk dipahami. Salah satu peran JQuery yaitu sebagai *javascript library* yang dapat memanggil fungsi yang terdapat di dalam *library* tersebut. JQuery diliris pertama kali pada tahun 2006 oleh Jhin Resig yang dapat mencuri perhatian para *developer web*. Hal ini dibuktikan oleh penggunaannya pada *website* seperti Google, Microsoft, Intel, Nokia, dan sebagainya.

Berikut beberapa kemampuan yang dimiliki oleh jQuery :

- Mempermudah akses dan memanipulasi elemen tertentu pada dokumen.
- Mempermudah modifikasi/perubahan tampilan halaman web.
- Mempersingkat Ajax (*Asynchronous Javascript and XML*)
- Memiliki API (*Application Programming Interface*)
- Mampu merespon interaksi antara user dengan halaman web lebih cepat.
- Menyediakan fasilitas untuk membuat animasi sekilas Flash dengan mudah.

Tanda yang digunakan dalam jQuery :

- Tanda “\$”, untuk mendefinisikan jQuery.
- (Selector), untuk menunjukkan elemen yang dipilih atau dituju.
- (Action), adalah jQuery action yang akan dilakukan terhadap elemen yang dipilih.

2.5 eXtensible Markup Language (XML)

eXtensible Markup Language (XML) adalah sebuah *markup language* yang menyediakan deskripsi, penyimpanan, dan format transmisi untuk pertukaran data melalui *webservices*. XML mirip dengan *HyperText Markup Language (HTML)* dimana keduanya memiliki elemen – elemen, attribute – attribute dan nilai – nilai. Perbedaan XML dan HTML yaitu XML didesain untuk mengirim dan menyimpan data dan berfokus kepada data apa yang dikirim atau disimpan, sedangkan HTML didesain untuk menampilkan data dan berfokus kepada bagaimana data tersebut ditampilkan [5]. Singkatnya, XML adalah tentang membawa informasi sedangkan HTML adalah tentang menampilkan informasi (Newcomer, 2002).

Sintaks XML yang digunakan dalam teknologi *web services* menspesifikasikan bagaimana data direpresentasikan, mendefinisikan bagaimana dan dengan kualitas layanan apa suatu data ditransmisikan, dan menjelaskan bagaimana *services* dipublikasikan dan ditemukan. Penggunaan XML dalam *web services* dibagi menjadi dua kategori yaitu sebuah representasi dan format penyimpanan data, dan sebuah spesifikasi dari *software* yang memanipulasi data. Meskipun XML semula merupakan sebuah *text markup language* untuk memproses dokumen, XML sering digunakan untuk memformat dan memanipulasi data.

Dokumen XML membentuk sebuah *tree structure* yang dimulai dari “the root” dan cabang – cabang menuju ke “the leaves”. Dokumen XML harus memiliki sebuah *root element* yang bertindak sebagai “the parent” dari semua elemen lain. Elemen – elemen dalam sebuah dokumen XML membentuk sebuah *document tree* yang dimulai dari *root*, *branches* dan menuju level terbawah dari *tree*. Semua elemen dapat memiliki subelemen atau elemen *child*. Istilah *parent*, *child* dan *sibling* digunakan untuk mendeskripsikan relasi antar elemen. *Parent* memiliki *children*, dan *children* pada level yang sama dinamakan *siblings* (*brothers* or *sisters*).

3. DESAIN SISTEM

Dalam sistem, terdapat tiga jenis aktor yaitu administrator, pengujian, dan siswa. Siswa adalah user yang dapat mengerjakan test secara online dan hanya dapat mengubah password, administrator adalah user yang dapat mengakses segala menu dan mengontrol adanya test, sedangkan pengujian adalah *user* yang memiliki batasan untuk mengakses menu tertentu dan pengujian

tersebut di inputkan oleh administrator terlebih dahulu sebelum pengujian melakukan *login*.

3.1 Database

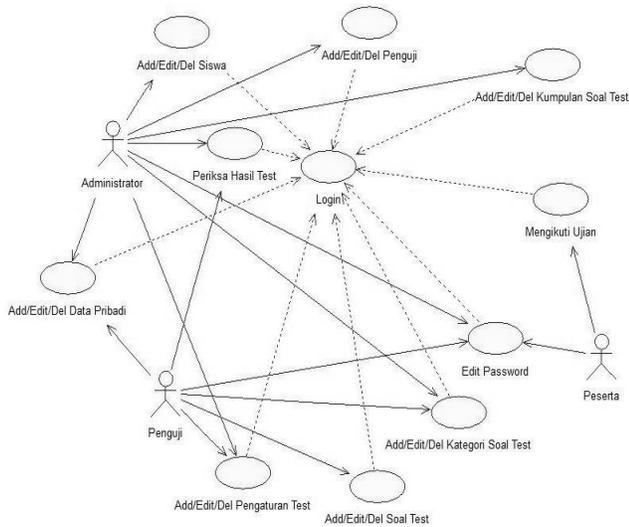
Desain *database* yang digunakan yaitu *database* pada *server* yang menggunakan *MySQL*. *Database* akan terhubung dengan *database server* sehingga segala fitur yang dilakukan akan langsung berhubungan dengan *database server*. *Use Case Diagram* yang digunakan dapat dilihat pada Gambar 2. *Desain Entity Relationship Diagram* dari *database server* dapat dilihat pada Gambar 3.

Dari Gambar 2 tentang *use case diagram* terlihat bahwa terdapat 11 *use case* yang terdiri dari 9 *use case* pada administrator, 6 *use case* pada pengujian, dan 2 *use case* pada peserta. Gambar 3 tentang *entity relationship diagram* terlihat bahwa terdapat 12 entitas dan 12 relasi yang terdiri dari 9 relasi *one to many* dan 3 relasi *one to one*.

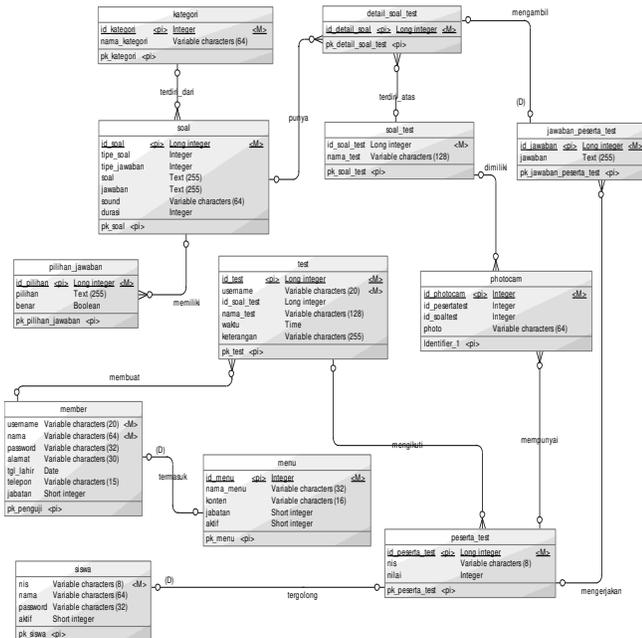
3.2 Web services

Web services yang dirancang menggunakan protokol SOAP dan berjumlah lima yaitu sebagai berikut.

- *Web service* untuk melakukan *login*. *Web service* ini terdiri dari satu *function* yaitu :
Function doLogin dengan parameter *login* dalam tipe data *string* dan dimasukkan ke array sehingga mengembalikan data *login* dalam tipe data *string* dalam bentuk pesan.
- *Web service* untuk menampilkan test yang tersedia dan telah diikuti oleh siswa. *Web service* ini terdiri dari satu *function* yaitu :
Function getScheduleTest dengan parameter *nis* dalam tipe data *string* dan dimasukkan ke array sehingga mengembalikan data *schedule test* dalam tipe data *string*.
- *Web service* untuk mengambil soal test dan menampilkannya. *Web service* ini terdiri dari dua *function* yaitu :
Function getSoalTest dengan parameter *param* berupa *id_peserta_test* dalam tipe data *string* dan mengembalikan data soal test dalam tipe data *string* pada array.



Gambar 2. Use Case Diagram

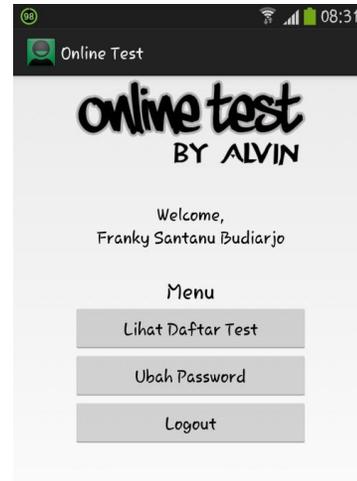


Gambar 3. Entity Relationship Diagram

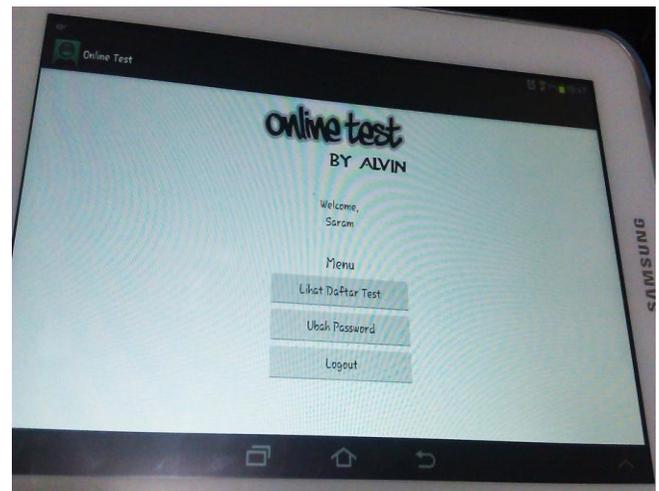
- *Web service* untuk mengirimkan jawaban ke database server. *Web service* ini terdiri dari satu *function* yaitu : *Function doPostJawaban* dengan paramater param berupa *id_peserta_test*, *id_soal*, dan jawaban dalam tipe data *string* dan mengembalikan status menjadi *true* ataupun mengembalikan *error message* dalam tipe data *string*.
- *Web service* untuk mengubah password. *Web service* ini terdiri dari satu *function* yaitu : *Function doUbahPassword* dengan paramater param dalam bentuk *nis* yang sedang *login* dalam tipe data *string* dan mengembalikan pesan konfirmasi keberhasilan mengubah password dalam tipe data *string*.

4. IMPLEMENTASI SISTEM

Aplikasi dimulai dengan menampilkan halaman *login* terlebih dahulu sebelum siswa, administrator, maupun penguji dapat mengakses menyunya. Setelah siswa melakukan *login*, maka dapat masuk ke menyunya yang terdiri dari *list* jadwal test, ubah password, maupun logout dari aplikasi ini seperti yang terlihat pada Gambar 4 yang merupakan Halaman Pengujian pada *screenshot* dari komputer, Gambar 5 yang merupakan Halaman Pengujian pada Samsung GT-P3100, dan Gambar 6 yang merupakan Halaman Pengujian pada Samsung I9500.



Gambar 4. Halaman Pengujian pada Screenshot dari komputer



Gambar 5. Halaman Pengujian pada Samsung GT-P3100



Gambar 6. Halaman Pengujian pada Samsung I9500

Ketika siswa telah siap untuk melakukan ujian, maka dapat mengklik nama test yang akan dikerjakannya sehingga siswa dapat masuk ke dalam halaman ujian. Halaman ujian ini memiliki durasi, dan soal dapat berupa audio, gambar, *multiple choice*, dan essay.

Sebelum administrator maupun penguji dapat mengakses menu, maka akan dilakukan *login* terlebih dahulu. *Login* tersebut berguna untuk memberi hak akses kepada administrator maupun penguji.

Setelah melakukan *login*, maka administrator dapat masuk ke dalam menu. Menu administrator terdiri dari 9 macam menu yaitu : data pribadi, input penguji, input siswa, input soal test, kategori soal test, kumpulan soal test, pengaturan soal test, periksa hasil test, dan ubah password. Penguji dapat mengakses menu yang sama, akan tetapi yang dapat membedakannya adalah tidak dapat mengakses menu input penguji dan input siswa.

Pada bagian *website* sebelum administrator dan penguji dapat masuk ke menu utama, maka harus melakukan *login* terlebih dahulu seperti yang terlihat pada Gambar 7. Halaman menu utama ini sama seperti pada menu di Android, akan tetapi yang membedakan koneksi internet yang didapatkan oleh *website* lebih cepat dari pada pada Android. Halaman menu utama pada *website* terdiri dari 9 macam menu yaitu :

Menu pertama dari halaman menu utama adalah data pribadi. Halaman ini berguna untuk menginputkan data pribadi administrator maupun penguji sehingga memudahkan untuk mengganti-gantinya. Akan tetapi hal yang dikhawatirkan yaitu terjadi ketidakcocokan antara data yang diinputkan oleh administrator maupun penguji dengan data pada database server yang telah tersimpan.

Menu kedua dari halaman menu utama adalah input penguji. Pada halaman ini, hanya administrator yang dapat mengakses menu ini dikarenakan administrator yang memiliki wewenang penuh terhadap jalannya test. Inputan pada halaman ini terdiri dari *username* dan nama lengkap penguji.

Menu ketiga dari halaman menu utama adalah input siswa. Halaman ini hanya dapat diakses oleh administrator saja. Halaman

ini berguna untuk menginputkan siswa yang akan mengikuti tes online. Siswa dapat melakukan *login* dengan password *default*.

Menu keempat dari halaman menu utama adalah input soal test. Halaman ini berguna untuk menginputkan soal test sehingga siswa dapat mengakses soal test yang akan di ujikan pada waktu tertentu dan jam tertentu. Jadwal untuk pengaturan test akan diatur oleh penguji maupun administrator. Pada halaman input soal, dapat membuat kategori terlebih dahulu sebelum membuat soal test. Halaman ini terdiri dari tipe soal, tipe jawaban, kategori soal, durasi, dan juga soal serta jawaban yang diinputkan oleh administrator maupun penguji untuk melakukan test.

Menu kelima dari halaman menu utama adalah kategori soal test. Halaman ini berguna untuk menginputkan kategori soal test sehingga dapat dengan mudah ditambahkan ke kumpulan soal dengan menggunakan filter yang tersedia. Sebelum menginputkan soal, maka administrator dan penguji harus menambahkan kategori soal test terlebih dahulu lalu menginputkan soal test. Halaman ini terdiri dari nama kategori soal test yang ingin dibuat sehingga mudah untuk mengelompokkannya.

Menu keenam dari halaman menu utama adalah kumpulan soal test. Halaman ini berguna untuk mempermudah administrator dan penguji dalam membuat test pada pengaturan test sehingga tidak membuang-buang waktu untuk membuat pengaturan test yang akan dilaksanakan. Ketika soal yang telah dipilih, maka dapat menentukan bobot soal kemudian setelah selesai, dapat menekan tombol update bobot soal.

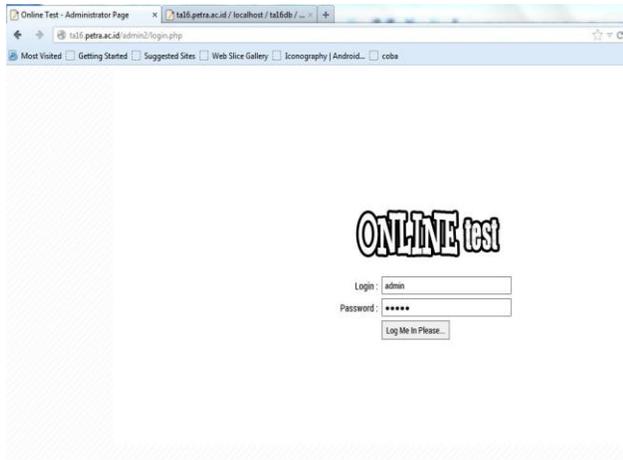
Menu ketujuh dari halaman menu utama adalah pengaturan test. Halaman ini berguna untuk mengatur test yang akan berlangsung dengan mengisi nama test, nama penguji, kumpulan test, waktu mulai, waktu selesai, menggunakan durasi atau pun tidak, dan keterangan. Peserta yang telah ditambahkan, maka akan hilang dari daftar yang dapat mengikuti test.

Menu kedelapan dari halaman menu utama adalah periksa hasil test. Halaman ini berguna untuk memeriksa hasil test yang telah dikerjakan oleh siswa sehingga penguji dapat dengan mudah untuk menilai hasil dari siswa kerjakan.

Menu kesembilan dari halaman menu utama adalah ubah password. Halaman ini berguna untuk mengubah password yang lama ke password yang baru sehingga tidak ada orang yang mengetahui password yang saat ini.

Halaman *login* siswa terdiri dari inputan nis dan juga password/ Inputan tersebut berguna untuk mencegah terjadinya masuk ke halaman menu siswa tanpa adanya inputan siswa yang dapat mengakses menu tersebut.

Setelah melakukan *login*, maka siswa akan melihat halaman *list* test. Halaman ini berguna untuk menampilkan test apa saja yang tersedia untuk siswa tertentu. Halaman ini berisikan nama test, waktu mulai, waktu selesai, status, dan nilai sehingga tidak terjadi ambiguitas dalam melihat test yang aktif untuk siswa.



Gambar 7. Halaman Login website administrator dan penguji

5. KESIMPULAN

Dari proses perancangan sistem hingga pengujian aplikasi yang telah dilakukan, dapat ditarik beberapa kesimpulan yaitu sebagai berikut.

- Pembagian hak akses untuk peserta, penguji dan administrator sangat penting. Hak akses dibagi agar masing-masing dapat mengakses menu yang sesuai. Menu tersebut digunakan untuk mempermudah mereka dalam menggunakan website ini.

- Berdasarkan hasil pengujian terhadap beberapa *Smartphone* dan tablet *device* dengan sistem operasi Android 2.3 (Gingerbread) dan 4.0 (Ice Cream Sandwich) dan 4.2.2 (Jelly Bean), aplikasi dapat berjalan dengan baik pada *device* dengan sistem operasi Android 4.0 (Ice Cream Sandwich) dan 4.2.2 (Jelly Bean). Namun, pada *device* dengan sistem operasi 2.3 (Gingerbread), akses *web service* hanya diijinkan terjadi pada foreground process. Hal ini bertentangan dengan versi 4.0 (Ice Cream Sandwich) sehingga akses *web service* pada *device* Android 2.3 (Gingerbread), tidak dapat dilakukan.

6. DAFTAR PUSTAKA

- [1] Hermawan, Stephanus S. (2011). *Mudah membuat aplikasi Android*. Yogyakarta : Andi.
- [2] Snell, James. (2001) *Programming Web services with SOAP*. United States : O'Reilly Media.
- [3] *WSDL and UDDI*. (1999). retrieved May 28, 2013 from http://www.w3schools.com/wSDL/wSDL_uddi.asp
- [4] Hakim, Lukmanul. (2010). *Bikin Website Super Keren dengan PHP dan jQuery*. Yogyakarta : Lokomedia.
- [5] *Introduction To XML*. (1999). retrieved May 22, 2013 from http://www.w3schools.com/xml/xml_whatIs.asp