

# APLIKASI BIRO JODOH BERBASIS WEB DAN J2ME

Kemas Faerusy Nizar<sup>1</sup>, Setiawardhana<sup>2</sup>, Idris Winarno<sup>3</sup>

Mahasiswa Jurusan Teknik Informatika<sup>1</sup>, Dosen Pembimbing<sup>2</sup>  
Politeknik Elektronika Negeri Surabaya  
Institut Teknologi Sepuluh Nopember  
Kampus PENS-ITS Keputih Sukolilo Surabaya 60111  
Telp. 031-5947280, 031-5946114, Fax: 031-5946114  
email: Nizar\_chie@yahoo.co.id  
Makalah Proyek Akhir

## ABSTRAK

Keberadaan Internet saat ini memberikan keuntungan secara langsung maupun tidak langsung kepada dunia bisnis, pendidikan, komunitas dan banyak lagi dari skala kecil hingga besar. Dengan adanya fasilitas internet data-data bisa disimpan, diambil dan dikirimkan secara mudah keseluruh penjuru. Sehingga tidak bisa dipungkiri keberadaan website internet menjadi kebutuhan utama dalam pemberian informasi tercepat di era globalisasi sekarang ini. Seiring dengan perkembangan teknologi komunikasi yang sangat cepat dewasa ini kami mencoba membangun sebuah layanan birojodoh yang sangat mudah diakses oleh member dimanapun, dengan menggunakan salah satu layanan yang disediakan oleh operator jaringan yaitu layanan GPRS dan sebuah WEB. Dengan adanya layanan birojodoh berbasis J2ME dan WEB berupa CMS (Content Management System) diharapkan dapat membantu pencari pasangan dalam mempertemukan dengan seseorang yang mereka cari berdasarkan kriteria-kriteria yang mereka inginkan.

Kata kunci: java, php, mysql

## I. PENDAHULUAN

### 1.1 LATAR BELAKANG

Proyek ini mencoba membangun suatu aplikasi Birojodoh berbasis J2ME. Aplikasi ini dibangun menggunakan bahasa pemrograman JAVA (J2ME) dan database MySQL, sedangkan pada aplikasi WEB menggunakan bahasa pemrograman PHP. Aplikasi birojodoh berbasis J2ME dan WEB ini diharapkan dapat membantu memberikan rekomendasi jodoh idaman kepada user atau member birojodoh berdasarkan kriteria-kriteria pasangan yang mereka inginkan.

Birojodoh adalah sebuah layanan yang membantu seseorang pencari pasangan dalam menemukan pasangan hidup, banyak hal atau aturan-aturan yang dibuat oleh layanan birojodoh tertentu untuk mempertemukan seorang member layanan tersebut pada pasangan yang didambakannya, namun seiring dengan perkembangan teknologi komunikasi yang sangat cepat dewasa ini kami mencoba membangun sebuah layanan birojodoh yang sangat mudah diakses oleh member dimanapun, dengan menggunakan salah satu layanan yang disediakan oleh operator jaringan yaitu layanan GPRS dan sebuah WEB. Dengan adanya layanan birojodoh berbasis J2ME dan WEB berupa CMS (Content Management System) diharapkan dapat membantu pencari pasangan dalam mempertemukan dengan seseorang yang mereka cari berdasarkan kriteria-kriteria yang mereka inginkan.

### 1.2 TUJUAN

Proyek akhir ini bertujuan untuk menyediakan fasilitas birojodoh bagi pencari jodoh yang berbasis J2ME dan web, sehingga akan memberikan layanan yang memudahkan bagi member untuk dapat bergabung

atau mendaftarkan diri sebagai member pasang jodoh atau pencari jodoh. Di aplikasi ini member dapat bergabung dengan mengirimkan kriteria dari pasangan yang dicari atau mengirimkan data dirinya untuk bergabung sebagai member pasang jodoh.

### 1.3 PERUMUSAN MASALAH

Bagaimana menyediakan fasilitas biro jodoh berbasis J2ME dan WEB untuk mendapatkan jodoh yang mereka inginkan berdasarkan sesuai kriteria-kriteria yang diinputkan oleh member birojodoh ini melalui Aplikasi J2ME ataupun WEB. Adapun permasalahan yang harus dihadapi antara lain :

1. Membuat satu database server untuk dua aplikasi yaitu aplikasi WEB dan aplikasi J2ME.
2. Mengimplementasikan hasil dari pengolahan data dari database ke dalam aplikasi J2ME.
3. Mengupload file pada aplikasi j2me.

### 1.4 BATASAN MASALAH

Batasan yang digunakan acuan pada pembuatan proyek akhir ini yaitu :

1. Aplikasi ini menggunakan bahasa pemrograman J2ME.
2. Web ini menggunakan bahasa pemrograman PHP.
3. Jenis handphone yang dipergunakan ialah handphone yang memiliki dukungan Java dan GPRS.

### 1.5 METODOLOGI

Dalam pengerjaan proyek akhir ini meliputi langkah-langkah sebagai berikut :

- **Studi Literatur**

Pada tahap ini akan diadakan studi literatur tentang biro jodoh, pemrograman WEB dan J2ME serta pengumpulan data pendukung yang dibutuhkan. Pada pengembangan aplikasi ini akan menggunakan *software Netbeans, XAMPP* sehingga dilakukan studi literatur tentang *software* ini.

- **Pengumpulan Data**

Data-data atau informasi yang diperoleh adalah data – data yang diambil dari internet

- **Perancangan Sistem**

Perencanaan pembuatan sistem meliputi perancangan sistem web.

- **Persiapan Data**

Data-data penunjang yang didapatkan berupa suatu kesimpulan, fakta-fakta dan aturan yang mengatur proses pencarian data yang saling berhubungan satu sama lain disimpan ke dalam basis data mysql sebagai media penyimpanan.

- **Pembuatan program komputer yang meliputi pembuatan antar muka dan mesin inferensi.**

Dalam pembuatan program, terlebih dahulu dibuat desain antar muka antara lain mengenai web dan lainnya.

- **Pengujian dan Analisa**

Pengujian dan analisa dimaksudkan untuk mengetahui sejauh mana sistem yang dibuat pada proyek akhir ini dapat berfungsi sesuai dengan proses sistem yang diharapkan.

- **Simpulan**

Dibuat kesimpulan dari pengujian sistem proyek akhir dengan membandingkan apakah hasilnya seperti yang diharapkan pada tujuan proyek akhir sebelumnya.

- **Pembuatan Laporan**

Membuat dokumentasi dari semua tahapan proses diatas berupa laporan yang berisi tentang dasar teori, hasil proyek akhir dan hasil analisa

## II. TEORI PENUNJANG

### 2.1 J2ME (Java 2 Micro Edition)

#### 2.1.1 J2ME Profile

Jika *J2ME Configuration* menyediakan *library-library* Java untuk implementasi fitur-fitur standar dari sebuah *handheld devices*, *J2ME Profile* menyediakan implementasi-implementasi tambahan yang sangat spesifik dari sebuah *handheld devices*. Saat ini ada lima kategori *J2ME Profile*, yakni:

- *Mobile Information Device Profile* (MIDP)
- *Foundation Profile* (FP)
- *Personal Profile*
- *RMI Profile*

*Personal Digital Assistance Profile*

#### 2.1.2 MIDP (Mobile Information Device Profile)

MIDP menyediakan *library-library* Java untuk implementasi dasar *interface* (GUI), implementasi jaringan (*networking*), *database*, dan *timer*. MIDP dirancang khusus untuk *wireless phone* dan *pager*.

#### 2.1.3 Arsitektur Aplikasi MIDlet

Paket *javax.microedition.midlet* memiliki kelas MIDlet yang mendefinisikan MIDP (*Mobile Information Device Profile*) dan interaksinya dengan perangkat handheld yang bersangkutan. Kelas ini merupakan kelas utama dalam sebuah MIDlet, yaitu sebuah aplikasi MIDlet harus merupakan turunan dari kelas ini. Kelas ini merupakan kelas abstrak, sehingga kelas turunannya harus mengimplementasikan fungsi-fungsi abstrak di dalamnya.

Perhatikan kode Java kelas MIDlet pada listing program berikut :

```
/*
 * Kelas Abstract MIDlet dalam paket
 * javax.microedition.midlet
 */
package javax.microedition.midlet;
import com.sun.midp.midlet.MIDletState;
import com.sun.midp.midlet.MIDletSuite;

public abstract class MIDlet{
    protected MIDlet(){
        state = new MIDletProxy(this);
    }
    protected abstract void startApp()
    throws MIDletStateChangeException;
    protected abstract void pauseApp();
    protected abstract void destroyApp(Boolean
    flag) throws MIDletStateChangeException;
    public final void notifyDestroyed(){
        state.notifyDestroyed();
    }
    public final void notifyPaused(){
        state.notifyPaused();
    }
    public final String getAppProperty(String Key){
        return.state.getMIDletSuite().getProperty(key);
    }
    public final void resumeRequest(){
        state.resumeRequest();
    }
    private MIDletProxy state;
}
```

#### 2.1.4 MIDP Record Store

Dalam perangkat *handphone* berbasis Java, disediakan media penyimpanan yang bersifat *non-volatile*. Artinya jika kita menyimpan pada media yang *non-volatile*, semacam *disket, hardisk*, data tersebut tidak akan hilang jika terjadi kehilangan energi listrik. Media penyimpanan *non-volatile* ini dalam MIDlet dikenal dengan nama RMS (*Record Management System*).

RMS merupakan implementasi database sederhana digunakan dalam MIDlet Java yang berorientasi record. Jika memanfaatkan fasilitas RMS ini, kita harus mengimport paket yang menyediakan fungsi-fungsi untuk manajemen RMS ini, yaitu *javax.microedition.rms.\**. Data yang ada di dalam database RMS ini dikenal dengan *RecordStore*.

Fungsi-fungsi dalam paket *javax.microedition.rms.\** menyediakan fungsi-fungsi untuk menghapus, menambah, mengubah data dan memungkinkan dua atau lebih MIDlet dalam satu aplikasi untuk berbagi data *RecordStore*.

*RecordStore* dapat memiliki nama unik dengan batasan 32 karakter dan bersifat *case sensitive*. Dalam satu project MIDlet kita tidak bisa membuat dua atau lebih *RecordStore*.

✓ **Manajemen Record Store**

Untuk membuka recordstore menggunakan fungsi statik.

```
public static RecordStore openRecordStore(String recordStoreName,
boolean createIfNecessary) throws
RecordStoreException,
RecordStoreFullException,
RecordStoreNotFoundException
```

Parameter *recordStoreName* adalah nama database yang akan dibuat, parameter *createIfNecessary* dapat bernilai *true/false*. Jika diset *true*, maka jika *recordstore* tidak ditemukan, otomatis akan langsung dibuat. Jika diset *false*, dan *recordstore* tidak ditemukan, maka melempar *RecordStoreNotFoundException*.

✓ **Menambah Data pada RecordStore**

Data dalam database RMS menggunakan *array of bytes*. Untuk menambah record baru menggunakan fungsi *addRecord()*.

```
public int addRecord(byte[] data, int offset, int numBytes) throws
RecordStoreNotOpenException,
RecordStoreException,
RecordStoreFullException.
```

Parameter :

byte[] data : *array* bit data yang akan ditambahkan ke *RecordStore*

int offset : index awal *array* data yang akan ditambahkan.

int numBytes : panjang bit data yang akan ditambahkan.

✓ **Mengambil Data dari RecordStore**

Untuk mengambil data dari *RecordStore*. Fungsi yang digunakan adalah *getRecord()*.

```
public byte[] getRecord(int RecordID) throws
RecordStoreNotOpenException,
InvalidRecordIDException,
RecordStoreException
```

RecordID : nomor RecordID yang akan diambil datanya.

✓ **Menghapus Data dari RecordStore.**

Untuk menghapus data dari *RecordStore*, fungsi yang digunakan adalah *deleteRecord(int RecordID)*

```
public void deleteRecord(int RecordID) throws
RecordStoreNotOpenException,
InvalidRecordIDException,
RecordStoreException
```

✓ **Mengganti Data dari RecordStore**

Untuk mengganti data dari *RecordStore*, fungsi yang

digunakan *setRecord()*.

```
public void setRecord(int RecordID, byte[] newData, int offset, int numBytes) throws
RecordStoreNotOpenException,
InvalidRecordIDException,
RecordStoreException,
RecordStoreFullException.
```

✓ **Mendapatkan Informasi RecordStore**

Setiap record ditandai dengan *RecordID*. Mencari *RecordID* berikutnya menggunakan fungsi *getNextRecordID()* dan untuk mengetahui jumlah record dengan fungsi *getNumRecords()*.

✓ **Menampilkan Daftar RecordStore.**

Untuk mendapatkan semua recordstore yang terdapat pada MIDlet dapat menggunakan fungsi *listRecordStores()*. Jika tidak ada *RecordStore* maka akan mengembalikan nilai null.

```
public static String[] listRecordStores()
```

✓ **Menghapus RecordStore**

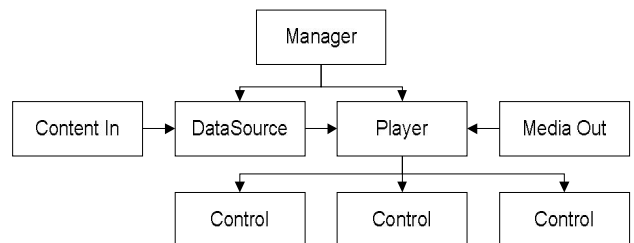
Untuk menghapus *RecordStore* yang ada dalam sistem, Anda bisa menggunakan fungsi berikut : *public void static deleteRecordStore(String name)*

Parameter name merupakan *RecordStore* yang akan dihapus.

**2.1.5 Akses Kamera**

MMAPI (mobile media API) adalah paket yang memungkinkan para pengembang MIDlet untuk dapat memainkan file multimedia di dalam perangkat telepon genggam dan juga melakukan kontrol terhadap kamera dalam pengambilan gambar. Perangkat yang digunakan untuk melakukan pekerjaan tersebut tentunya harus mendukung infrastruktur atau prasarana yang dibutuhkan serta dukungan terhadap tipe file media yang akan dimainkan.

MMAPI dirancang untuk mendukung beragam tipe file media yang dapat dimainkan di dalam perangkat telepon genggam, dan juga untuk menyediakan mekanisme pengambilan data (data capturing). MMAPI memfasilitasi sebuah kelas factoru generik yang menginstansi objek-objek media palyer dari lokasi tertentu (URI, objek-objek *InputStream*, maupun objek-objek *DataSource*). Sedangkan implementasi dari perangkat telepon genggam yang mendukung MMAPI menyediakan kelas player konkrit untuk media-media yang didukung. Player dari perangkat tersebut memiliki fitur spesifik dand apat dibuka melalu interface *Control*. Sebagai contoh video player akan memiliki *VideoControl* dan *RecordControl*. Interaksi dengan player dari sebuah telepon genggam dilakukan melalui control-kontrolnya.



**Gambar 2.4.** Arsitektur MMAPI

Pada intinya, untuk membuat aplikasi multimedia di dalam telepon genggam, terlebih dahulu harus membuat instances dari objek Player berdasarkan sumber data (DataSource) yang ditentukan. Setelah objek Player terbantu, maka dapat membuat objek-objek control melalui objek Player yang ada

Kelas Manager adalah kelas factory generic static di dalam MMAPi yang digunakan untuk menginstansi objek Player. Untuk membuat instance Player perlu memanggil metode createPlayer() dari kelas Manajer.

Jenis method createPlayer() yang paling umum digunakan adalah yang memiliki parameter bertipe String yaitu URI locator, untuk image capture URL memerlukan format string khusus yaitu capture://video . mode video akan terus menampilkan video dari kamera sampai memerintahkan untuk mengambil gambar (capture).

Untuk akses kamera dan pengambilan gambar yang dilakukan pertama kali adalah melewati parameter capture://video pada saat pemanggilan method createPlayer ()

```
Player player;  
Player=Manager.createPlayer("capture://video");
```

selanjutnya player diinisialisasi dengan memanggil method realized

```
player.realized
```

selain itu perlu menggunakan control VideoControl sebagai penampil video yang akan diambil gambarnya, atau dikenal dengan istilah view Finder

```
VideoControl videocontrol;  
Videocontrol= (VideoControl) player.getControl  
("VideoControl");
```

videoControl memiliki dua mode display dalam hal menampilkan gambar, yaitu USE\_DIRECT\_VIDEO (dengan menggunakan canvas) dan USE\_PRIMITIVE\_GUI (dengan menggunakan form). Dalam contoh ini, akan menggunakan objek canvas untuk menampilkan gambar kamera. Mode display atau tampilan tersebut dapat diset melalui pemanggilan method initDisplayMode() dari objek VideoControl

```
videocontrol.initDisplayMode(USE_DIRECTI_VIDEO,ca  
nvas);
```

dalam hal ini canvas adalah objek yang akan dibuat untuk menampilkan gambar kamera. Sampai di sini, sudah dapat menjalankan atau mengaktifkan kamera dalam telepon genggam, yaitu dengan memanggil method start()

```
player.start();
```

selanjutnya, setelah player aktif, dapat mengambil gambar yang ditampilkan oleh kamera dengan menggunakan

method getSnapshot() yang disediakan oleh objek VideoControl/ gambar yang diambil method getSnapshot() akan disimpan dalam bentuk array byte

```
byte[] imageData=videocontrol.getSnapshot(null);
```

parameter dari method getSnapshot() adalah berupa string yang menunjukkan format gambar yang akan digunakan , seperti tipe file (PNG,JPEG, atau yang lainnya) serta lebar serta tinggi gambar yang akan diambil, nilai null yang dilewatkan ke dalam method getSnapshot() diatas menandakan bahwa data yang diambil akan disimpan dalam format PNG dengan ukuran default 160 X 120 yang kemungkinan besar dapat didukung oleh semua tipe telepon genggam.

gambar hasil foto kemudian akan diproses lebih lanjut, misalnya disimpan atau dikirim lewat koneksi internet. Untuk menjadikan objek image dari array byte dapat dilakukan dengan perintah berikut

```
Image  
image=Image.createImage(imageData,0,imageData.lenght);
```

dengan pembuatan objek image maka hasil gambar bisa ditampilkan di form atau canvas [8]

### 2.1.5 HTTP (HyperText Transfer Protocol)

HTTP (HyperText Transfer Protocol) adalah protokol yang dipergunakan untuk mentransfer dokumen dalam World Wide Web (WWW). Protokol ini adalah protokol ringan, tidak berstatus dan generik yang dapat dipergunakan berbagai macam tipe dokumen.

Pengembangan HTTP dikoordinasi oleh Konsorsium World Wide Web (W3C) dan grup bekerja Internet Engineering Task Force (IETF), bekerja dalam publikasi satu seri RFC, yang paling terkenal RFC 2616, yang menjelaskan HTTP/1.1, versi HTTP yang digunakan umum sekarang ini.

HTTP adalah sebuah protokol meminta/menjawab antara client dan server. Sebuah client HTTP seperti web browser, biasanya memulai permintaan dengan membuat hubungan TCP/IP ke port tertentu di tuan rumah yang jauh (biasanya port 80). Sebuah server HTTP yang mendengarkan di port tersebut menunggu client mengirim kode permintaan (request), seperti "GET / HTTP/1.1" (yang akan meminta halaman yang sudah ditentukan), diikuti dengan pesan MIME yang memiliki beberapa informasi kode kepala yang menjelaskan aspek dari permintaan tersebut, diikuti dengan badan dari data tertentu.

Beberapa kepala (header) juga bebas ditulis atau tidak, sementara lainnya (seperti tuan rumah) diperlukan oleh protokol HTTP/1.1. Begitu menerima kode permintaan (dan pesan, bila ada), server mengirim kembali kode jawaban, seperti "200 OK", dan sebuah pesan yang diminta, atau sebuah pesan error atau pesan lainnya.

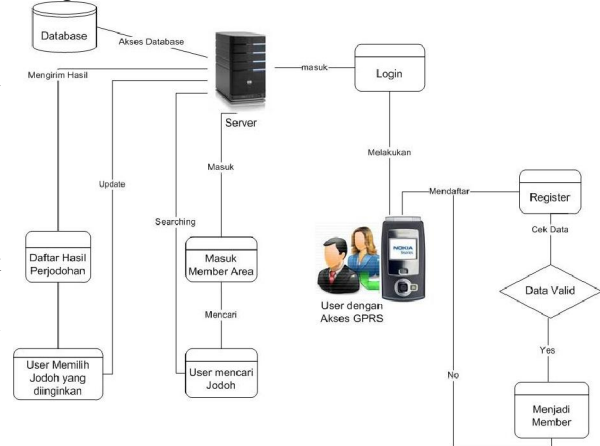
### 2.1.5 Pengertian PHP

Pertama kali PHP diperkenalkan oleh Rasmus Lerdorf (1995) menggunakan nama PHP/FI (Personal HomePage/Form Interpreter), yang kemudian muncul PHP 3.0 yang

diperkenalkan oleh Andi Gutmans dan Zeev Suraski pada tahun 1997. PHP banyak digunakan oleh programmer berlatar belakang C/C++ karena kemiripan syntaxnya. Open source, karenanya gratis dan bebas. Database pasangannya biasanya MySQL, dijalankan bersama webserver Apache di atas operating system Linux. Semuanya gratis dan bebas. Saat ini PHP sudah sampai versi 5.0.

PHP adalah suatu bahasa pemrograman web yang digunakan untuk keperluan CGI (Common GatewayInterface), artinya mempunyai kemampuan untuk membaca variabel dari client dan mengirimkan ke server untuk kemudian di server diolah dan hasilnya dikembalikan ke client. PHP adalah pemrograman yang bersifat Server Side, artinya program hanya bisa diakses melalui server, sedangkan client tidak bisa mengakses program PHP.

perancangan system yang dibuat pada tugas akhir ini adalah sebagai berikut:



Gambar 3.1 Perancangan System

## 2.2 J2ME

Pada proyek akhir ini J2ME yang digunakan adalah akses kamera, capture picture dan J2ME networking, yaitu menghubungkan perangkat telepon genggam ke internet.

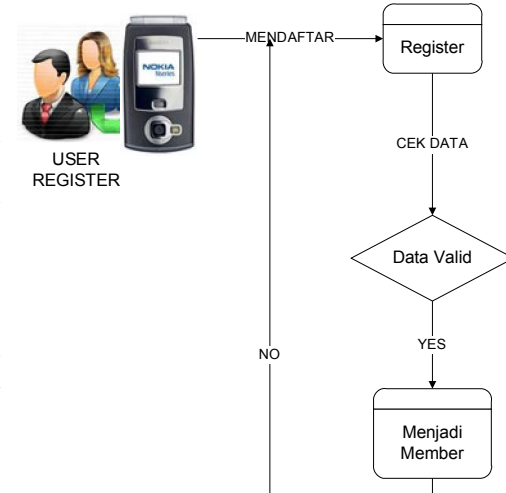
Untuk akses kamera menggunakan paket MMAPI (Mobile Media API) dengan membuat kelas player dengan menggunakan video dan pengontrolannya menggunakan videoControl gambar yang dihasilkan adalah hasil dari snapshot gambar video Class Connector adalah inti dari framework networking, karena semua koneksi jaringan hanya dapat dibuat oleh class Connector dengan static methodnya yaitu open(). Berbagai koneksi jaringan dapat dibuat dengan method yang sama dengan cara mengirimkan parameter yang berbeda. Pada aplikasi proyek akhir ini koneksi yang dibangun adalah http, yaitu mengakses webserver. Untuk membuat koneksi http, butuh class HttpURLConnection, class ini berasal dari ContentConnection, sehingga bisa menangani stream yang berasal dari InputConnection dan OutputConnection [8]

## 3.1.1 Perencanaan Sistem

Proses awal adalah user mendaftarkan sebagai member pada aplikasi ini melalui web maupun mobile :

## 3.2 PERANCANGAN SISTEM

- Proses Register :



Gambar 3.2 Proses Register

## III. PERANCANGAN DAN IMPLEMENTASI SISTEM

### 3.1 PERANCANGAN SISTEM

Dalam kondisi sekarang ini banyak layanan biro jodoh, layanan ini diperlukan untuk menemukan pasangan hidup sesuai dengan kriteria yang dicari. Namun perlu adanya kecepatan dalam menyampaikan informasi-informasi layanan tersebut.

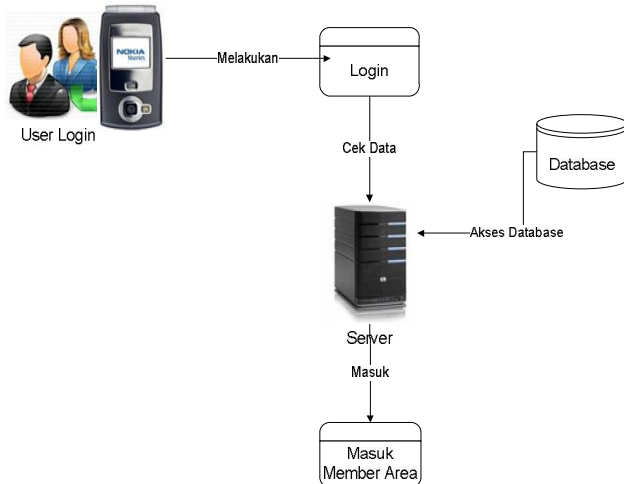
Untuk itulah dibuatnya aplikasi web dan mobile untuk memudahkan user dalam menemukan pasangan hidupnya, dimana aplikasi mobile yang menggunakan bahasa pemrograman j2me dan aplikasi web menggunakan bahasa pemrograman php. Yang mana, informasi yang akan ditampilkan kepada pengguna adalah berupa hasil proses dari user.

Didalam bab ini akan dibahas mengenai perencanaan dan pembuatan Aplikasi secara keseluruhan,

Penjelasan :

- User melakukan pendaftaran melalui web maupun aplikasi mobile, lalu system akan melakukan cek data.
- Jika data yang dimasukkan benar atau valid user akan menjadi member dalam aplikasi ini.
- Dan jika user salah memasukkan data yang diminta oleh system user tidak dapat menjadi member dan user akan melakukan proses dari awal yaitu memasukkan data.

- Proses Login :

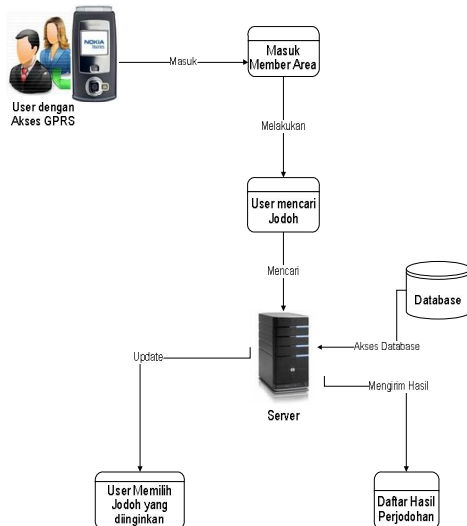


**Gambar 3.3** Proses Login

Penjelasan :

- User melakukan *Login* setelah melakukan pendaftaran atau *registrasi* melalui web atau *mobile*.
- Server akan melempar *inputan* user ke database server, apabila data yang dimasukkan user sesuai dengan database, user akan memasuki member area atau menu profil dari user itu sendiri.
- User melakukan *Login* setelah itu user melakukan pencarian jodoh berdasarkan kriteria jodoh yang di inginkan user.

- Proses Mencari Jodoh :



**Gambar 3.4** Proses Mencari Jodoh

- Penjelasan :

- Server akan menampilkan semua user yang telah menjadi member, setelah melakukan pencarian jodoh.
- Jodoh yang di inginkan akan masuk ke dalam Daftar Hasil Perjodohan atau *Member List*.

## IV. ANALISA DAN UJI COBA HASIL SISTEM

Uji coba sistem dilakukan cara melalui emulator dengan web diletakkan di server computer local.

### 4.1 Pengujian pada web

Pada halaman awal web, user mengisi kolom – kolom *register* untuk menjadi member aktif, jika yang diisikan user benar maka user telah menjadi member, namun user harus mengaktifasi email yang telah di isikan agar dapat menjadi member aktif.

Setelah memasukkan *username* dan *password* di form *login* dengan benar maka user telah masuk ke dalam halaman member. Dalam halaman member terdapat beberapa menu yang tersedia.

Setelah berada di dalam member area, user dapat memilih berapa menu yang tersedia disana. Salah satunya menu *member list*. Dalam menu *member list* kita dapat melihat user lain yang telah menjadi member aktif dalam aplikasi ini, dan kita dapat melihat profil lengkapnya dengan jelas.

Dalam halaman profil lain tersebut terdapat tombol *bookmark*, fungsinya yaitu memasukkan ke dalam halaman *bookmark* kita untuk menjadi daftar teman kita, berikut adalah halaman *bookmark* dimana member yang sebelumnya telah kita lihat profilnya dan kita *bookmark* masuk ke dalam halaman *bookmark* kita.

Di dalam member area juga terdapat menu profil, dimana dalam menu profil tersebut kita dapat melihat profil kita sendiri.

Dalam halaman profil terdapat tombol *edit profil*, dimana kita dapat merubah profil diri kita sendiri, seperti gambar di bawah ini. Setelah melakukan perubahan profil kita menekan tombol *update*, maka profil kita akan diperbaharui, jika ingin membatalkan tekan tombol batal yang telah tersedia.

Dan yang terakhir adalah menu *logout*, jika masuk ke dalam menu ini maka member akan keluar dari halaman member area.

### 4.2 Pengujian pada mobile

Terdapat beberapa menu di halaman pertama pada aplikasi *mobile*. Dalam menu *register*, user memasukkan data-data ke dalam *form* yang telah di sediakan oleh sistem.

Setelah user melakukan *registrasi*, maka user telah menjadi member dan dapat melakukan *login*. Setelah berhasil *login*, member masuk ke dalam member area yang di dalamnya terdapat beberapa menu..

Pada menu *View Profile*, kita dapat melihat profil diri kita sendiri, Pada menu *view profile* juga terdapat menu *edit* dan menu *back*, menu *edit* digunakan untuk memperbaharui

profil sedangkan menu back digunakan untuk kembali ke halaman sebelumnya.

Pada menu *edit* kita dapat merubah profil diri kita, dan Pada menu sebelumnya juga terdapat menu *member list* yang dimana dalam menu *member list* terdapat member aktif dalam aplikasi ini.

Pada menu *member list* di dalamnya terdapat beberapa menu yaitu menu back, member selanjutnya, dan menu jadi teman, pada menu member selanjutnya, kita dapat melihat member siapa saja yang aktif dalam aplikasi ini, pada menu jadi teman kita dapat memasukkan daftar member untuk jadi teman, kemudian akan masuk kedalam halaman *bookmark* kita.

Pada menu dalam member area juga terdapat menu *upload foto*, diman untuk mengirimkan gambar ke dalam server aplikasi ini, jika kita menekan tombol *Upload Foto*, secara langsung kamera akan menyala, jika kita menekan tombol ambil, akan mendapatkan hasil foto yang diinginkan, dan akan keluar hasil fotonya.

## V. KESIMPULAN DAN SARAN

### 5.1 KESIMPULAN

Setelah dilakukan pengujian dan analisa program, maka dapat diperoleh kesimpulan sebagai berikut :

- dari hasil percobaan, menunjukkan aplikasi ini mampu digunakan untuk mencari jodoh yang di inginkan member berdasarkan inputan yang di masukkan pada waktu registrasi.
- sistem mampu melakukan pembaharuan profil setelah menjadi member aktif.

### 5.2 SARAN

Dari beberapa simpulan yang telah diambil, maka dapat dikemukakan saran-saran yang akan sangat membantu untuk pengembangan aplikasi ini selanjutnya.

- Perlu diadakan penambahan data untuk inputan registrasi, dengan cara menambahkan form pada kolom registrasi.
- Untuk aplikasi mobile diharapkan bisa lebih kompatibel di semua tipe telepon genggam.

## VI. DAFTAR PUSTAKA

- [1] [www.gealgeol.com/2008/02/12/mysql.html](http://www.gealgeol.com/2008/02/12/mysql.html)
- [2] [http://www.wirelessdevnet.com/channels/java/features/j2me\\_http.phtml](http://www.wirelessdevnet.com/channels/java/features/j2me_http.phtml)
- [3] Budi Raharjo, *Tuntunan Pemrograman Java untuk Handphone* Bandung. Informatika : 2007
- [4] M Sholahuddin, *Pemrograman J2ME belajar cepat pemrograman perangkat telekomunikasi mobile*. Bandung. Informatika : 2006
- [5] modul kuliah pemrograman lanjut, PENS ITS
- [6] Endy Muhardin, "PHP Programming Fundamental dan MySQL Fundamental", ArtiVisi Intermedia, 2003.

[7] Nana Ramadijanti, "Aplikasi Web Dasar Pemrograman PHP", Politeknik Elektronika Negeri Surabaya, Surabaya 2008.

[8] Lukmanul Hakim, "Membongkar Trik rahasia Para Master PHP", LOKOMEDIA, Yogyakarta 2008.