

**APLIKASI POHON KELUARGA (SILSILAH) MENGGUNAKAN
BASISDATA RELASIONAL**

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat
Untuk Memperoleh Gelar Sarjana
Jurusan Teknik Informatika



Oleh:

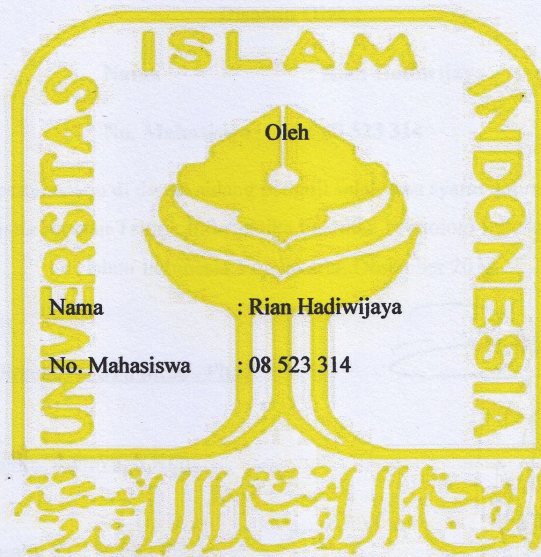
Nama : Rian Hadiwijaya

No. Mahasiswa : 08 523 314

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA**

2014

LEMBAR PENGESAHAN
APLIKASI POHON KELUARGA (SILSILAH) MENGGUNAKAN
BASISDATA RELASIONAL

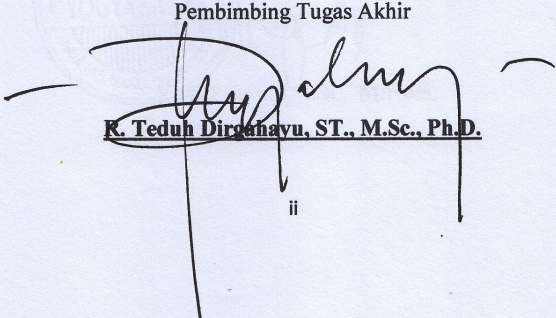


Oleh
Nama : Rian Hadiwijaya
No. Mahasiswa : 08 523 314

Yogyakarta, Februari 2014

Menyetujui,

Pembimbing Tugas Akhir


R. Teduh Dirgantayu, ST., M.Sc., Ph.D.

LEMBAR PENGESAHAN DOSEN PENGUJI
APLIKASI POHON KELUARGA (SILSILAH) MENGGUNAKAN
BASISDATA RELASIONAL

Disusun Oleh :

Nama : Rian Hadiwijaya

No. Mahasiswa : 08 523 314

Telah dipertahankan di depan sidang penguji salah satu syarat untuk memperoleh
gelar sarjana Jurusan Teknik Informatika Fakultas Teknologi Industri Universitas
Islam Indonesia Yogyakarta, Desember 2013

Tim Penguji,

R. Teduh Dirgahayu, ST., M.Sc., Ph.D.

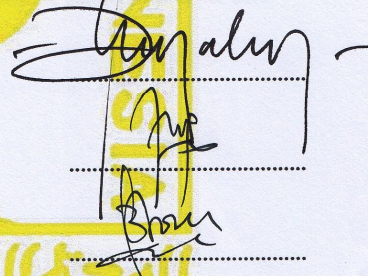
Ketua

Nur Wijayaning Rahayu, S.Kom.

Anggota I

Manik Hapsara, ST., M.Sc., Ph.D.

Anggota II

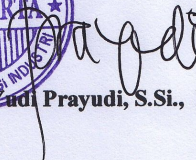


Mengetahui,

Ketua Jurusan Teknik Informatika
Fakultas Teknologi Industri Universitas Islam Indonesia



Yadi Prayudi, S.Si., M.Kom.



HALAMAN PERSEMBAHAN

Dengan mengucapkan syukur Alhamdulillah, kupersembahkan karya kecilku ini untuk orang-orang yang kusayangi :

- Ayah dan bunda tercinta, motivator terbesar dalam hidupku yang tak pernah jemu mendo'akn dan menyayangiku, atas semua pengorbanan dan kesabaran mengantarku sampai saat ini. Tak pernah cukup ku membalas cinta dan kasih dari Ayah dan Bunda kepadaku
- Saudariku Eriani Dwi Rahmawati
- Sahabat-sahabat seperjuangan sewaktu masa kuliah di Teknik Informatika Universitas Islam Indonesia yang tidak bisa penulis sebutkan satu per satu.

HALAMAN MOTTO

" Kegagalan hanya terjadi bila kita menyerah " (Lessing)

" Manusia tidak merancang untuk gagal, meraka gagal untuk merancang "

(William J. Siegel)"

" Hai orang-orang yang beriman, Jadikanlah sabar dan shalatmu Sebagai penolongmu, sesungguhnya Allah beserta orang-orang yang sabar"

(Al-Baqarah: 153)

" Do What You Love, Love What You Do "

KATA PENGANTAR



Assalamu'alaikum Warahmatullahi Wabarakatuh

Alhamdulillah, puji syukur kita panjatkan atas kehadiran Allah SWT. berkat rahmat dan karuniaNya penulis bisa menyelesaikan tugas akhir ini. Tugas akhir ini diajukan sebagai salah satu syarat untuk memperoleh gelar sarjana di Jurusan Teknik Informatika Universitas Islam Indonesia.

Dalam penyelesaian tugas akhir ini, penulis menyadari tidak lepas dari dorongan semangat dan do'a dari berbagai pihak, sehingga tugas akhir ini dapat diselesaikan. Semoga Allah SWT. membalas kebaikan kepada semua pihak yang telah mendukung. Selanjutnya, perkenankan penulis untuk menyampaikan terimakasih kepada:

1. Allah SWT, karena atas rahmat dan karuniaNya lah penulis bisa dan mampu menyelesaikan tugas akhir ini.
2. Nabi Muhammad SAW. sebagai junjungan umat islam.
3. Ayahanda Sahadi, S.Pd. dan Ibunda Risnawati Hartani, S.Sip., yang telah memberikan dukungan dan do'a kepada penulis dalam menyelesaikan tugas akhir ini.
4. Eriani Dwi Rahmawati, adinda dari penulis.
5. Bapak Dr. H. Eddy Suandi Hamid, M.Ec., selaku rektor Universitas Islam Indonesia.
6. Bapak Ir. Gumbolo Hadi Susanto, M.Sc., selaku dekan Fakultas Teknologi Industri Universitas Islam Indonesia.
7. Bapak Yusuf Yudi Prayudi, S.Si., M.Kom., selaku ketua jurusan Teknik Informatika Universitas Islam Indonesia.

8. Bapak R. Teduh Dirgahayu, ST., M.Sc., P.Hd. selaku dosen pembimbing yang telah meluangkan waktunya ditengah kesibukan. Terimakasih atas bimbingan dan masukan yang telah diberikan serta dukungan dalam pengerjaan tugas akhir ini.
9. Seluruh dosen Teknik Informatika Universitas Islam Indonesia yang selama ini telah berbagi ilmu selama masa kuliah.
10. Seluruh karyawan di Program Studi Teknik Informatika Universitas Islam Indonesia, terimakasih atas segala pelayanan dan bantuannya kepada penulis selama ini.
11. Teman-teman seperjuangan selama masa kuliah Budi Satriya, Andri Prabowo, Catur Romadhoni, Wahyu Santoso, terimakasih atas semua dukungan dan bantuannya kepada penulis.
12. Teman-teman kontrakan, Lelsi, Mbah So, Zen, Taufik, terimakasih telah memberi semangat dalam menyelesaikan tugas akhir ini.

Penulis Menyadari bahwa laporan tugas akhir ini masih jauh dari kesempurnaan. Untuk itu penulis menerima saran dan kritik yang membangun untuk menyempurnakan Aplikasi Pohon Keluarga(Silsilah) Menggunakan Basisdata Relasional. Semoga dapat menambah wawasan dan ilmu serta bermanfaat bagi masyarakat luas.

Wassalamu'alaikum Warahmatullahi Wabarakatuh

Yogyakarta, Februari 2014

Penulis

SARI

Silsilah keluarga merupakan satu hal yang mungkin sering dianggap kurang penting, dikarenakan bagi sebagian orang merasa malas dalam mencari garis keturunannya. Dengan demikian, dikembangkan sebuah aplikasi yang dapat mendokumentasikan sebuah silsilah keluarga.

Aplikasi ini dapat mendokumentasikan silsilah keluarga seseorang yang telah dimasukkan kedalam basisdata. Dalam mendokumentasikan silsilah keluarga ini, digunakan basisdata relasional untuk mendapatkan hubungan kekeluargaannya. Dalam aplikasi ini juga terdapat sebuah fasilitas untuk mengecek keturunan dan mengecek moyang dari seseorang.

Untuk dapat mendokumentasikan silsilah tersebut, aplikasi ini menggunakan metode *Depth-First Search*, dimana akan menelusuri pada satu cabang hingga mencapai akhir dari cabang tersebut, kemudian akan mencari pada cabang berikutnya.

Kata Kunci : *Silsilah, Database Relasional, Tree, Struktur Data*

TAKARIR

<i>admin</i>	: Orang-orang yang bertugas untuk mengurus hal-hal administrasi
<i>backtracking</i>	: Merupakan salah satu algoritma yang didasarkan pada pencarian ruang solusi
<i>database</i>	: Wadah untuk menyimpan data
<i>dropdown</i>	: Sebuah menu yang memanjang ke bawah
<i>flowchart</i>	: Sebuah diagram dengan simbol-simbol grafis yang menyatakan aliran algoritma
<i>foreign key</i>	: Sebuah kunci asing dalam database relasional
<i>hyperlink</i>	: Acuan dalam dokumen ke dokumen yang lain
<i>input</i>	: Memasukkan data
<i>login</i>	: Proses untuk masuk ke dalam sebuah sistem
<i>primary key</i>	: Kunci utama dalam sebuah basis data
<i>source code</i>	: Kumpulan kode-kode aplikasi
<i>spreadsheet</i>	: Tabel empat persegi (atau garis-garis) yang terdiri dari informasi tertentu.

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PENGESAHAN DOSEN PEMBIMBING	ii
LEMBAR PENGESAHAN DOSEN PENGUJI	iii
HALAMAN PERSEMBAHAN	iv
HALAMAN MOTTO	v
KATA PENGANTAR	vi
SARI	viii
TAKARIR	ix
DAFTAR ISI	x
DAFTAR TABEL	xiii
DAFTAR GAMBAR	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	2
1.5 Manfaat Penelitian	2
1.6 Metodologi Penelitian	2

1.7 Sitematika Penulisan	4
BAB II LANDASAN TEORI	5
2.1 Struktur Data.....	5
2.1.1 Struktur Data Tree (pohon)	6
2.2 Basis Data	7
2.2.1 Tipe-Tipe Basis Data	8
2.2.2 Model Basis Data Relasional	8
BAB III METODOLOGI.....	16
3.1 Studi Pustaka.....	16
3.2 Pengembangan Aplikasi	16
3.2.1 Analisis Sistem	16
3.2.2 Perancangan Sistem	16
3.2.2.1 Desain Antarmuka Sistem	17
3.2.2.2 Perancangan Basisdata	18
3.2.2.3 <i>Flowchart</i>	24
3.2.3 Implementasi Sistem	28
3.3 Skenario Pengujian Sistem	28
BAB IV HASIL DAN PEMBAHASAN	31
4.1 Implementasi.....	31

4.1.1 Halaman Muka	31
4.1.2 Halaman Login	32
4.1.3 Halaman Cari Data	33
4.1.4 Halaman Cek Keturunan	33
4.2 Pembahasan	34
4.2.1 Pengelolaan Data	34
4.2.2 Pencarian Data.....	42
4.3 Kelebihan dan Kekurangan Sistem	49
4.3.1 Kelebihan Sistem	49
4.3.2 Kekurangan Sistem	50
BAB V KESIMPULAN DAN SARAN	51
5.1 Kesimpulan	51
5.2 Saran	51
DAFTAR PUSTAKA	52
LAMPIRAN	53

DAFTAR TABEL

Tabel 3.1 Struktur Awal Tabel Personal	19
Tabel 3.2 Struktur Awal Tabel Keluarga	20
Tabel 3.3 Struktur Awal Tabel Admin	20
Tabel 3.4 Struktur Tabel Personal	21
Tabel 3.5 Struktur Tabel Keluarga	22
Tabel 3.6 Struktur Tabel Ayah	22
Tabel 3.7 Struktur Tabel Ibu	23
Tabel 3.8 Struktur Tabel Anak	23
Tabel 3.9 Struktur Tabel Admin.....	23
Tabel 3.10 Skenario Pengujian Sistem	29
Tabel 4.1 Hasil Pengujian Sistem	47

DAFTAR GAMBAR

Gambar 3.1 Desain Antarmuka Halaman Utama	17
Gambar 3.2 Desain Antarmuka Halaman Login Admin	18
Gambar 3.3 Desain Antarmuka Halaman Pencarian	18
Gambar 3.4 Desain Antarmuka Halaman Cek Keturunan	19
Gambar 3.5 <i>Entity Relationship Diagram</i> (ERD)	24
Gambar 3.6 <i>Flowchart</i> Proses Pencarian Keturunan	25
Gambar 3.7 <i>Flowchart</i> Pengecekan Keturunan	26
Gambar 3.8 <i>Flowchart</i> Proses Pencarian Moyang	27
Gambar 3.9 <i>Flowchart</i> Proses Pengecekan Moyang	27
Gambar 3.10 Data Pohon Keluarga Sebagai Data Uji Sistem	28
Gambar 4.1 Tampilan Halaman Muka Untuk Pengguna Biasa	31
Gambar 4.2 Tampilan Halaman Muka Untuk Admin	32
Gambar 4.3 Tampilan Halaman <i>Login</i> Admin	32
Gambar 4.4 Tampilan Halaman Cari Data.....	33
Gambar 4.5 Tampilan Halaman Cek Keturunan	34
Gambar 4.6 Formulir Tambah Data Personal	35
Gambar 4.7 Tampilan Halaman Personal	35
Gambar 4.8 Tampilan Halaman Detail	36

Gambar 4.9 Tampilan Halaman Detail Yang Sudah Meninggal.....	37
Gambar 4.10 Tampilan Halaman Edit Personal	38
Gambar 4.11 Formulir Tambah Data Keluarga.....	39
Gambar 4.12 Formulir Tambah Keluarga Lanjutan	39
Gambar 4.13 Tampilan Halaman Keluarga.....	40
Gambar 4.14 Tampilan Halaman Edit Keluarga	40
Gambar 4.15 Tampilan Halaman Detail Keluarga	41
Gambar 4.16 Formulir Tambah Data Ayah	41
Gambar 4.17 Formulir Tambah Data Ibu.....	42
Gambar 4.18 Formulir Tambah Data Anak	42
Gambar 4.19 Tampilan Menu Untuk Pengguna Biasa	43
Gambar 4.20 Tampilan Menu Untuk Admin	43
Gambar 4.21 Tampilan Halaman Cari Data.....	43
Gambar 4.22 Hasil Dari Pencarian Keturunan	44
Gambar 4.23 Menu <i>Dropdown</i> Cek Keturunan	45
Gambar 4.24 Formulir Untuk Pengecekan Keturunan	45
Gambar 4.25 Hasil Dari Pengecekan Keturunan	45
Gambar 4.26 Hasil Dari Pencarian Moyang	46
Gambar 4.27 Hasil Dari Pengecekan Moyang	46

BAB I

PENDAHULUAN

1.1 Latar Belakang

Hubungan kekeluargaan atau kekerabatan merupakan hubungan antara tiap entitas yang memiliki asal-usul silsilah yang sama, baik melalui keturunan biologi, sosial, maupun budaya. Dalam antropologi, sistem kekerabatan termasuk keturunan dan pernikahan, sementara dalam biologi istilah ini termasuk keturunan dan perkawinan. Hubungan kekerabatan manusia melalui pernikahan umum disebut sebagai "hubungan dekat" daripada "keturunan" (juga disebut "konsanguitas"), meskipun kedua hal itu bisa tumpang tindih dalam pernikahan di antara orang-orang yang satu moyang. Hubungan kekeluargaan ini sebagaimana genealogi budaya dapat ditarik kembali pada Tuhan.

Genealogi adalah kajian tentang keluarga dan penelusuran jalur keturunan serta sejarahnya. Ahli genealogi menggunakan berita dari mulut ke mulut, catatan sejarah, analisis genetik, serta rekaman lain untuk mendapatkan informasi mengenai suatu keluarga dan menunjukkan kekerabatan dan silsilah dari anggota-anggotanya. Hasilnya sering ditampilkan dalam bentuk bagan (disebut bagan silsilah) atau ditulis dalam bentuk narasi.

Perangkat lunak genealogi ini adalah salah satu perangkat lunak yang digunakan untuk mempermudah dalam membuat silsilah keluarga. Keuntungan menggunakan perangkat lunak ini adalah garis keturunan seseorang yang sudah terdokumentasi bisa dilihat secara detail.

Dokumentasi hubungan kekeluargaan ini dapat digunakan untuk mencari hubungan kekeluargaan seseorang. Dengan hal tersebut, seseorang dapat mengetahui silsilah keluarganya menggunakan perangkat lunak ini. Data istilah pada perangkat lunak ini dapat disimpan dengan memanfaatkan basisdata relasional.

Basisdata relasional adalah tipe basisdata yang paling populer digunakan saat ini. Banyak perkembangan yang telah dibuat dari model basisdata sebelumnya menyederhanakan manajemen data dan pengambilan data. Data menjadi lebih mudah diatur, terutama melalui struktur tabel relasional dan meminta data tertentu tanpa perlu memahami struktur basisdata.

Ada beberapa keuntungan dari basisdata relasional ini, diantaranya :

- a. Data dapat diakses secara cepat.
- b. Struktur basisdata mudah diubah.
- c. Pengguna mudah membuat query yang kompleks untuk mengambil data.
- d. Pengguna mudah menerapkan integristas data.
- e. Pengguna mudah membuat dan memodifikasi program aplikasi.
- f. Bahasa standar (SQL) sudah dibuat.

1.2 Rumusan Masalah

Dalam hal ini, rumusan masalahnya adalah bagaimana membangun sebuah aplikasi yang dapat menyimpan atau mendokumentasikan keturunan maupun moyang seseorang, sehingga nantinya dapat diketahui siapa saja keturunan maupun moyangnya.

1.3 Batasan Masalah

Untuk lebih memfokuskan ruang lingkup pengembangannya, maka dibentuk batasan masalah nya adalah sebagai berikut :

1. Program ini berbasis web sederhana
2. Menggunakan basis data MySQL sebagai media penyimpanan data
3. Pengisian data (*input*) hanya dapat dilakukan oleh *Admin*
4. Menggunakan konsep basisdata relasional

1.4 Tujuan Penelitian

Membangun sebuah aplikasi yang berbasis web dimana nantinya dapat digunakan untuk mencari sebuah garis keturunan antar seseorang dan mendokumentasikan sebuah hubungan kekeluargaan pada sebuah keluarga.

1.5 Manfaat Penelitian

Beberapa manfaat dari penelitian ini, antara lain :

- a. Dapat memahami lebih lanjut tentang basisdata relasional.
- b. Dapat memahami lebih lanjut pengaplikasian basisdata relasional dalam sebuah website.
- c. Bagaimana penerapan fungsi yang bersifat rekursif.

1.6 Metodologi Penelitian

Metodologi yang akan digunakan dalam penelitian ini adalah :

1. Melakukan studi pustaka.
Studi kepustakaan perlu untuk dilakukan untuk menambah pengetahuan dan wawasan terhadap apa yang sedang diteliti.
2. Melakukan tahap pengembangan aplikasi.
Pada tahapan ini ada beberapa hal yang akan dilakuka, antara lain :
 - a. Analisis sistem.
 - b. Desain sistem.
 - c. Implementasi sistem
3. Pengujian sistem.
Setelah implementasi selesai dan dianggap sempurna, maka aplikasi ini akan diuji sesuai dengan fungsi dan tujuannya.

1.7 Sistematika Penulisan

Dalam sistematika penulisan laporan tugas akhir ini dibagi dalam beberapa bab, sebagai berikut :

Bab 1 Pendahuluan, bab ini membahas secara umum tentang tugas akhir yang terdiri dari latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian serta sistematika penulisan.

Bab 2 Landasan Teori, pada bab ini membahas beberapa teori yang terkait dengan basis data relasional dan masalah yang dibahas.

Bab 3 Metodologi, pada bagian ini membahas uraian-uraian dan langkah-langkah dalam penyelesaian masalah.

Bab 4 Hasil dan Pembahasan, pada bab ini membahas secara rinci hasil dari perancangan terhadap aplikasi yang dibuat serta segala sesuatu yang berkaitan dengan hal tersebut.

Bab 5 Kesimpulan dan Saran, yaitu kesimpulan dari hasil penelitian yang telah dilakukan serta saran-saran yang dianggap perlu.

BAB II

LANDASAN TEORI

Aplikasi ini nantinya akan menggunakan sebuah pohon keluarga yang berbentuk tree. Dengan pohon keluarga berbentuk tree tersebut nantinya akan dikonversi kedalam bentuk tabel yang memiliki relasi antar tabel untuk menjaga setiap hubungan kekeluargaan seperti pada pohon keluarga yang berbentuk tree. Namun penggunaan relasi antar tabel ini nantinya, hanya menggunakan ruang lingkup yang sangat kecil dalam keluarga. Dalam artian hanya menggunakan keluarga kecil yang terdiri dari ayah, ibu dan anak. Dengan demikian pada saat peroses pencariannya, dengan menggunakan hal tersebut dapat menampilkan yang tidak jauh berbeda dengan pohon keluarga yang berbentuk tree.

Struktur data berbentuk pohon/tree adalah di mana data serta hubungan antar data direpresentasikan dengan record dan link (pointer), dimana record-record tersebut disusun dalam bentuk tree (pohon), dan masing-masing node pada tree tersebut merupakan record/grup data elemen dan memiliki hubungan cardinalitas 1:1 dan 1:M, sedangkan relasional berbentuk tabel, di mana data serta hubungan antar data direpresentasikan oleh sejumlah tabel dan masing-masing tabel terdiri dari beberapa kolom yang namanya *unique*.

2.1 Struktur Data

Menurut Teddy Marcus Zakaria dan Agus Prijono (2006) struktur data adalah model logika/matematik yang secara khusus mengorganisasikan data. Sebuah model harus dapat mencerminkan kethubungan data dengan dunia nyata dan bentuknya sederhana. Terdapat dua jenis struktur data, yaitu :

- a. Struktur data statis, yaitu struktur yang tidak berubah seperti array, rekord, himpunan.
- b. Struktur data dinamis, yaitu struktur yang berubah seperti list, queue(antrian), stack(tumpukan).

Beberapa jenis struktur data telah disediakan pada bahasa pemrograman(contohnya bahasa Pascal), yaitu struktur data Array, Set, Record, String. Sedangkan struktur data lainnya yang belum ada dapat dibentuk sendiri misalnya struktur data List, Stack, Queue, Tree, Graph.

Penelitian ini lebih lanjut akan memfokuskan penggunaan struktur data yang berbentuk Tree. Struktur data tree(pohon) adalah struktur data yang diproses melalui alamat akarnya(root). Struktur data ini merupakan struktur yang dinamis, di mana ukurannya boleh berubah. Data-data pada struktur data ini mempunyai hubungan hirarki antar elemennya.

2.1.1 Struktur Data Tree (pohon)

Sebuah pohon adalah struktur data yang terdiri dari akar (root), dan subpohon-subpohon dalam susunan berhirarki. Suatu pohon dapat bersifat statik dan dinamik. Pohon statik adalah keadaan di mana bentuk pohon sudah ditentukan. Sedangkan pohon dinamik adalah keadaan di mana bentuk pohon dapat berubah karena operasi penambahan dan penghapusan simpul (node).

Dalam struktur data tree ini terdapat beberapa istilah-istilah dasar, antara lain :

- a. Simpul/Node/Vertex : merupakan elemen-elemen pohon yang mengandung informasi atau data dan penunjuk percabangan.
- b. Akar/Root : node yang menjadi pangkal.
- c. Tingkat/Level :tingkat atau level dari akar (*root*) didefinisikan sebagai 0, sedangkan tingkat dari simpul lainnya didefinisikan sebagai 0 + tingkat dari orangtua dari simpul tersebut.
- d. Kedalaman : kedalaman (*depth*) dari suatu pohon dinyatakan oleh tingkat yang tertinggi dari simpul yang terdapat pada pohon tersebut.
- e. Derajat dari simpul : jumlah anak yang dimiliki oleh sebuah simpul.

- f. Daun : simpul paling luar (node eksternal) atau simpul yang tidak memiliki anak (berderajat 0).
- g. Simpul Nonterminal : disebut juga node internal atau simpul yang memiliki anak (berderajat tidak 0).

Dalam sebuah struktur data berbentuk tree, ada beberapa algoritma pencarian, antara lain :

a. *Depth First Search* (DFS)

Pada algoritma DFS, pencarian dilakukan pada satu node dalam setiap level dari yang paling kiri. Jika pada level yang paling dalam solusi belum ditemukan, maka pencarian dilanjutkan pada node sebelah kanan. Node yang di kiri dapat dihapus dari memori. Jika pada level yang paling dalam belum ditemukan solusi, maka pencarian dilanjutkan ke level sebelumnya. Demikian seterusnya sampai ditemukannya solusi. Jika solusi ditemukan, maka tidak diperlukan proses backtracking (penelusuran untuk mendapatkan jalur yang diinginkan). Beberapa kelebihan dari algoritma DFS adalah pemakaian memori hanya sedikit karena hanya menyimpan lintasan yang aktif saja. Selain itu kelebihannya adalah jika solusi berada pada level yang paling dalam dan paling kiri, maka DFS akan menemukannya secara cepat.

2.2 Basisdata

Menurut Stephens dan Plew (2000) basisdata adalah mekanisme yang digunakan untuk menyimpan informasi atau data. Informasi adalah sesuatu yang kita gunakan sehari-hari untuk berbagai alasan. Dengan basisdata, pengguna dapat menyimpan data secara terorganisasi.

Kemudian, menurut Silberchatz, dkk. (2002) mendefinisikan basisdata sebagai kumpulan data berisi informasi yang sesuai untuk sebuah perusahaan. Sistem manajemen basisdata (DBMS) adalah kumpulan data yang saling berhubungan dan kumpulan program untuk mengakses data. Tujuan utama sistem

manajemen basisdata adalah menyediakan cara menyimpan dan mengambil informasi basisdata secara mudah dan efisien. Ramakrishnan dan Gehrke(2003) menyatakan basisdata sebagai kumpulan data, umumnya mendeskripsikan aktivitas satu organisasi atau lebih yang berhubungan.

Definisi basisdata, menurut McLeod, dkk.(2001) adalah kumpulan seluruh sumber daya berbasis komputer milik organisasi. Sistem manajemen basisdata adalah aplikasi perangkat lunak yang menyimpan struktur basisdata, hubungan antar data dalam basisdata, serta berbagai formulir dan laporan yang berkaitan dengan basisdata.

2.2.1 Tipe-Tipe Basisdata

Ada beberapa tipe basisdata. Berikut adalah tipe-tipe basisdata :

- a. Model basisdata file (*Flat file database model*)
- b. Model basisdata hierarki (*Hierarchical database model*)
- c. Model basisdata jaringan (*Network database model*)
- d. Model basisdata relasional (*Relational database model*)
- e. Model basisdata berorientasi objek (*Object oriented database model*)
- f. Model basisdata relasi objek (*Object relational database model*)

Penelitian ini akan lebih berfokus pada model basisdata relasional. Sehingga akan dibahas secara mendalam tentang basisdata relasional itu sendiri.

2.2.2 Model Basisdata Relasional

Basisdata relasional adalah "...tipe basisdata yang paling populer digunakan saat ini. Banyak perkembangan yang telah dibuat dari model basisdata sebelumnya menyederhanakan manajemen data dan pengambilan data. Data menjadi lebih mudah diatur, terutama melalui struktur tabel relasional dan meminta data tertentu tanpa perlu memahami struktur basisdata...".(Simarmata dan Iman,2006:27).

Dalam basisdata relasional terdapat beberapa macam istilah antara lain tabel, kolom, baris, field, normalisasi, integritas data, bahasa SQL. Berikut akan dijelaskan sedikit tentang istilah-istilah tersebut:

a. Tabel

Tabel adalah unit penyimpanan fisik utama data dalam basisdata. Dalam sebuah basisdata, bisa dikatakan tabel adalah sesuatu yang penting. Ketika mengakses basisdata, pengguna akan mengacu pada tabel untuk data yang diinginkan. Secara umum, basisdata bisa terdiri dari banyak tabel.

b. Kolom

Kategori informasi yang terdapat dalam tabel. Kolom bagi tabel seperti atribut bagi entitas. Dengan kata lain, ketika model bisnis diubah menjadi model basisdata, entitas menjadi tabel dan atribut menjadi kolom.

c. Baris

Satu baris data adalah kumpulan semua kolom yang berhubungan kejadian tunggal. Dengan kata lain, satu baris data adalah satu record dalam tabel.

d. Field

Sebuah field merupakan irisan antara sebuah baris dan sebuah kolom. Istilah field mirip dengan "sel" dalam program *spreadsheet*. Satu field hanya berisi satu data/nilai. Bisa jadi sebuah field tidak berisi data/nilai apapun. Kondisi ini disebut dengan nilai null.

e. Normalisasi Tabel

"...Normalisasi adalah teknik perancangan yang banyak digunakan sebagai pemandu dalam merancang basisdata relasional. Pada dasarnya, normalisasi adalah proses dua langkah meletakkan data dalam bentuk tabulasi dengan menghilangkan kelompok berulang lalu menghilangkan data yang terduplikasi dari tabel relasional. Tanpa normalisasi, sistem basisdata menjadi tidak akurat, lambat, tidak

efisien, serta tidak memberikan data yang di harapkan...". (Simarmata dan Iman,2006:80)

Pada waktu normalisasi basisdata, ada empat tujuan yang harus dicapai, yaitu:

1. Mengatur data dalam kelompok-kelompok sehingga masing-masing kelompok hanya menangani bagian kecil sistem.
2. Meminimalkan jumlah data berulang dalam basisdata.
3. Membuat basisdatayang datanya diakses dan dimanipulasi secara cepat dan efisien tanpa melupakan integritas data.
4. Mengatur data sedemikian rupa sehingga ketika memodifikasi data, hanya mengubah pada satu tempat.

Perancang basisdata terkadang menyebut keempat tujuan dengan istilah integritas data, integritas refrensial, dan pengaksesan data. Tujuan normalisasi adalah membuat kumpulan tabel relasional yang bebas dari data berulang dan dapat dimodifikasi secara benar dan konsisten. Ini berarti bahwa semua tabel pada basisdata relasional harus berada pada bentuk normal ketiga(3NF). Berikut adalah penjelasan mengenai normalisasi tabel pada basisdata relasional :

1. Bentuk normal kesatu (1nf)

Sebuah tabel relasional secara definisi selalu berada dalam bentuk normal pertama.

2. Bentuk normal kedua (2NF)

Definisi bentuk normal kedua menyatakan bahawa tabel dengan kunci utama gabungan hanya dapat berada pada 1NF.

3. Bentuk normal ketiga (3NF)

Bentuk normal ketiga mengharuskan semua kolom pada tabel relasional tergantung hanya pada kunci utama. Secara definisi, sebuah tabel berada pada bentuk normal ketiga (3NF) jika tabel sudah berada pada 2NF dan setiap kolom yang bukan kunci tidak tergantung secara transitif pada kunci utamanya.

4. Bentuk normal Boyce-Code (BCNF)

Setelah 3NF, semua masalah normalisasi hanya melibatkan tabel yang mempunyai tiga kolom atau lebih dan semua kolom adalah kunci. Bentuk normal boyce-code (BCNF) adalah versi 3NF yang lebih teliti dan berhubungan dengan tabel relasional yang mempunyai (a) banyak kunci kandidat, (b) kunci kandidat gabungan, dan (c) kunci kandidat yang saling tumpang tindih.

5. Bentuk normal keempat (4NF)

Sebuah tabel relasional berada pada bentuk normal keempat (4NF) jika dia dalam BCNF dan semua ketergantungan multivalued merupakan ketergantungan fungsional. Bentuk normal keempat didasarkan pada konsep ketergantungan multivalued (MVD).

6. Bentuk normal kelima (5NF)

Sebuah tabel berada pada bentuk normal kelima (5NF) jika dia tidak dapat mempunyai dekomposisi *lossless* menjadi sejumlah tabel lebih kecil. Empat bentuk normal pertama berdasarkan pada konsep ketergantungan fungsional, sedangkan bentuk normal kelima berdasarkan pada konsep ketergantungan gabungan (*join dependence*). Ketergantungan gabungan berarti bahwa sebuah tabel, setelah didekomposisi menjadi tiga atau lebih tabel yang lebih kecil, harus dapat digabungkan kembali untuk membentuk tabel asal.

f. Integritas Data

Integritas data adalah jaminan konsistensi data terhadap semua status konstrain yang diberlakukan terhadap data tersebut, sehingga memberikan jaminan keabsahan data itu sendiri. Integritas data sangat erat kaitannya dengan keamanan atau keberadaan data, dimana dapat terjadi secara institusional atau asidental. Terdapat beberapa jenis integritas data, sebagai berikut:

1. Entity Integritas

Entity integritas mendefinisikan sebuah baris sebagai sebuah entitas yang unik suatu tabel. Entity integritas memaksa integritas

dari kolom atau *primary key* dari suatu tabel, dimana sebuah *primary key* tidak boleh null.

2. Domain Integritas

Domain integritas merupakan validasi dari masukan sebuah kolom. Seorang DBA dapat memaksa domain integritas dengan membatasi tipe, format, atau range nilai-nilai yang mungkin (melalui *foreign key*, constraints, check constraints, default definition dan rules).

3. Refrential Integritas

Refrential integritas memastikan bahwa seluruh nilai dari *foreign key* cocok dengan nilai *primary key* yang dihubungkan.

4. User Defined Integritas

User defined integritas mengizinkan untuk menentukan spesifik *business rules* sendiri yang tidak sama pada kategori yang lain.

g. Perintah (statament) SQL

Sistem basisdata menyediakan bahasa pendefinisian data (*Data Definition Language-DDL*) untuk menentukan skema dan bahasa manupulasi data (*Data Manipukation Language-DML*) untuk menyatakan query dan update basisdata. Pada praktiknya, DDL dan DML bukan merupakan dua bahasa yang terpisah melainkan membentuk bagian bahasa basisdata, seperti yang umum digunakan pada bahasa SQL.

"...SQL adalah bahasa komputer standar ANSI (*American National Standard Institute*) untuk mengakses dan memanipulasi sistem basisdata. Pernyataan SQL digunakan untuk mengambil dan meng-update data dalam basisdata...". (Simarmata dan Iman, 2006:107)

SQL adalah kependekan dari *Structure Query Language* yang merupakan bahasa basisdata standar industri. SQL menyediakan *Data Definition Language* (DDL) dan *Data Manipulation Language* (DML). Walaupun dalam beberapa bagian saling berkaitan, perintah DDL memungkinkan untuk membuat dan mendefinisikan tabel

(CREATE TABLE), dan indeks (CREATE INDEX). Sebaliknya, perintah DML memungkinkan kita memanipulasi data dan membangun query untuk mengambil (SELECT) beberapa tabel, menyisipkan (INSERT) data baru, memperbaiki (UPDATE) data, dan menghapus (DELETE) data.

Bahasa SQL mengikutsertakan sintaks untuk update, insert, dan delete record. Perintah query dan update secara bersama membentuk bahasa manipulasi data SQL.

a. SELECT.

Pernyataan SELECT digunakan untuk memilih data dari tabel. Hasilnya disimpan dalam tabel hasil yang disebut Result Set.

- Distinct
Untuk memilih hanya nilai yang berbeda.
- Klausa WHERE
Klausa WHERE digunakan untuk menentukan kriteria seleksi. Untuk memilih data suatu tabel dengan kriteria tertentu, klausa WHERE dapat ditambahkan pada pernyataan SELECT.
- Kondisi LIKE
Kondisi LIKE digunakan untuk menentukan pencarian berdasarkan pola tertentu pada suatu kolom.

b. UPDATE

Pernyataan UPDATE digunakan untuk memodifikasi data dalam tabel.

c. DELETE

Pernyataan DELETE digunakan untuk menghapus baris pada tabel.

d. INSERT INTO

Pernyataan INSERT INTO digunakan untuk memasukkan data baru pada tabel.

Bagian bahasa definisi data dari SQL memungkinkan tabel basisdata dibuat atau dihapus.

a. CREATE TABLE

Pernyataan CREATE TABLE digunakan untuk membuat sebuah tabel basisdata baru.

b. ALTER TABLE

Pernyataan ALTER TABLE digunakan untuk mengubah tabel basisdata.

c. DROP TABLE

Pernyataan DROP TABLE digunakan untuk menghapus tabel basisdata.

d. CREATE INDEX

Pernyataan CREATE INDEX digunakan untuk membuat sebuah indeks.

e. DROP INDEX

Pernyataan DROP INDEX digunakan untuk menghapus sebuah indeks.

Berikut beberapa keuntungan model basisdata relasional adalah :

- a. Data dapat diakses secara cepat.
- b. Struktur basisdata mudah diubah.
- c. Pengguna mudah membuat query yang kompleks untuk mengambil data.
- d. Pengguna mudah menerapkan integristas data.
- e. Pengguna mudah membuat dan memodifikasi program aplikasi.
- f. Bahasa standar (SQL) sudah dibuat.

Berikut beberapa kekurangan basisdata relasional adalah :

- a. Kelompok informasi atau tabel yang berbeda harus dihubungkan untuk mengambil data.

- b. Pengguna harus memahami hubungan antar tabel.
- c. Pengguna harus belajar SQL.

BAB III

METODOLOGI

3.1 Studi Pustaka

Studi pustaka perlu untuk dilakukan dikarenakan untuk menambah wawasan peneliti tentang apa yang sedang diteliti. Sumber pustaka didapatkan dari buku maupun catatan atau informasi dari internet. Studi kepustakaan ini bertujuan untuk menghimpun informasi yang relevan dengan topik atau masalah yang sedang diteliti, sehingga sedikit tidak dapat menyempurnakan apa yang sedang diteliti.

3.2 Pengembangan aplikasi

Setelah melakukan studi pustaka, tahap selanjutnya adalah melakukan pengembangan aplikasi. Pada tahapan ini ada beberapa hal yang akan dilakukan, antara lain :

3.2.1 Analisis sistem

Analisis sistem bertujuan untuk mengidentifikasi permasalahan-permasalahan yang ada pada aplikasi. Segala permasalahan yang telah dikumpulkan akan dianalisis dan kemudian dapat menentukan alur dan tahap perancangan.

3.2.2 Perancangan sistem

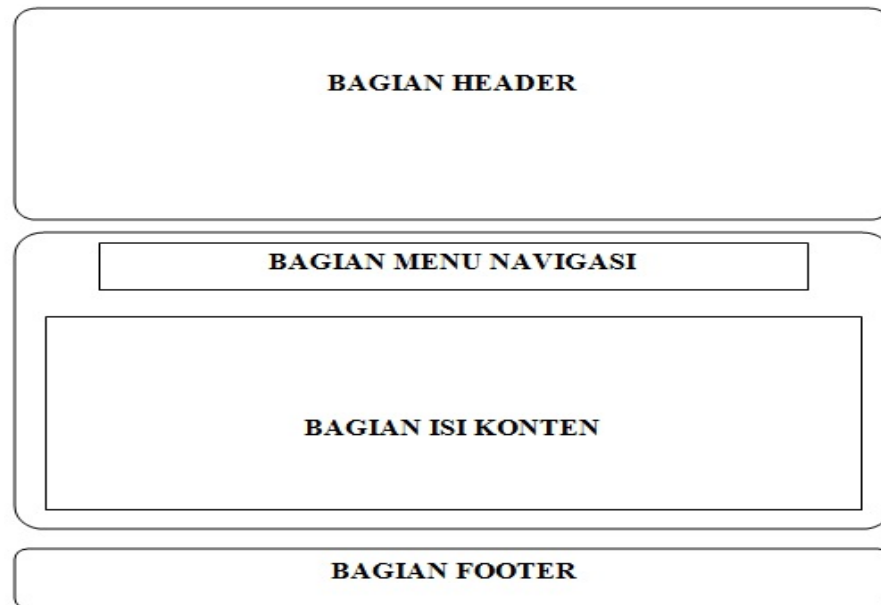
Sebelum melakukan pengimplementasian sebuah sistem maupun perangkat lunak, perlu adanya sebuah perancangan. Dalam perancangan aplikasi ini berdasarkan apa yang ditemukan selama proses analisis sistem. Dari hasil tersebut akan ditransformasikan ke dalam bentuk diagram dan desain antarmuka sebagai dasar dalam pengembangan selanjutnya.

3.2.2.1 Desain antarmuka sistem

Setiap sistem perlu sebuah desain untuk mendapatkan hasil yang sesuai dengan apa yang diinginkan. Berikut beberapa desain antarmuka dari aplikasi pohon keluarga (silsilah) menggunakan basisdata relasional.

- a. Desain antarmuka halaman utama.

Berikut adalah rancangan antarmuka halaman utama dari aplikasi pohon keluarga (silsilah) menggunakan basisdata relasional.



Gambar 3.1 Desain Antarmuka Halaman Utama

Desain halaman muka tersebut berlaku untuk halaman admin dan halaman untuk pengguna biasa. Perbedaannya terdapat pada bagian menu navigasi, di mana menu navigasi untuk halaman admin akan lebih banyak dikarenakan digunakan untuk manajemen data. Sedangkan untuk halaman utama pengguna biasa tidak terdapat menu untuk manajemen data.

- b. Desain antarmuka halaman login admin.

Sebelum seorang admin masuk ke halaman utama admin, dia akan melakukan *login* terlebih dahulu. Hal ini untuk menjaga agar sebuah sistem tidak disalahgunakan oleh pihak yang tidak

bertanggungjawab. Desain antarmuka halaman *login* untuk admin terlihat seperti pada gambar 3.2 berikut ini.

```

classDiagram
    class LoginAdmin {
        Username : 
        Password : 
        Login
    }
  
```

Gambar 3.2 Desain Antarmuka Halaman *Login* Admin

- c. Desain antarmuka halaman cari data.

Aplikasi ini sudah dilengkapi dengan fasilitas pencarian data, sehingga pengguna lebih mudah mencari data yang diinginkan. Gambar 3.3 berikut adalah desain antarmuka untuk halaman

```

classDiagram
    class SearchPage {
        BAGIAN HEADER
        BAGIAN MENU NAVIGASI
        BAGIAN HASIL PENCARIAN
        BAGIAN FOOTER
    }
  
```

Gambar 3.3 Desain Antarmuka Halaman Pencarian

- d. Desain antarmuka halaman cek keturunan.

Gambar 3.4 berikut adalah desain antarmuka untuk halaman cek

BAGIAN HEADER

BAGIAN MENU NAVIGASI

CEK KETURUNAN

Nama Moyang

Nama Keturunan

BAGIAN FOOTER

Gambar 3.4 Desain Antarmuka Halaman Cek Keturunan

3.2.2.2 Perancangan Basis data

Basis data merupakan suatu hal yang penting dalam penelitian ini. Basis data digunakan sebagai media penyimpanan data yang akan digunakan pada sistem. Semua data yang dibutuhkan akan disimpan pada basis data ini, dan nantinya akan digunakan kembali sesuai dengan kebutuhan dari sistem. Dalam perancangan basis data ini terdapat dua macam, yakni yang pertama adalah perancang untuk struktur tabel dan yang kedua adalah perancangan untuk hubungan antar tabel yang akan digambarkan menggunakan *Entity Relationship Diagram*(ERD). Berikut adalah rancangan awal untuk tabel yang akan digunakan.

Tabel 3.1 Struktur Awal Tabel Personal

FIELD	TYPE	SIZE	KETERANGAN
id	Integer	3	<i>Primary Key</i> (id personal)
nama	Varchar	50	Nama Lengkap

nama_alias	Varchar	50	
tgl_lahir	Varchar	2	
bulan	Varchar	20	
tahun	Varchar	4	
jenis_kelamin	Varchar	15	
alamat	Varchar	300	
pekerjaan	Varchar	50	
nikah	Varchar	15	Status Pernikahan
no_tlp	Varchar	20	
status	Varchar	50	Meninggal atau Masih Hidup
tgl_m	Varchar	2	Tanggal meninggal
bulan_m	Varchar	20	Bulan meninggal
tahun_m	Varchar	4	Tahun meninggal
tempat_makam	Varchar	50	Tempat dimakamkan
foto	Varchar	50	

Tabel 3.2 Struktur Awal Tabel Keluarga

FIELD	TYPE	SIZE	KETERANGAN
id_keluarga	Integer	3	<i>Primary Key</i> (id personal)
id_ayah	Integer	3	<i>Foreign Key</i> (id personal)
id_ibu	Integer	3	<i>Foreign Key</i> (id personal)
id_anak	Integer	3	<i>Foreign Key</i> (id personal)

Tabel 3.3 Struktur Awal Tabel Admin

FIELD	TYPE	SIZE	KETERANGAN
id_admin	Integer	3	<i>Primary Key</i>
uname	varchar	20	
paswd	Integer	100	Password

Dengan melihat struktur tabel di atas, maka dikhawatirkan adalah terjadinya redudansi data untuk setiap nama ayah dan nama ibu yang memiliki jumlah anak lebih dari satu. Dengan demikian, dilakukan normalisasi tabel, guna meminimalkan atau mengurangi kemungkinan terjadinya redudansi data. Setelah dilakukan normalisasi sedemikian rupa untuk menghilangkan atau mengurangi kemungkinan terjadi untuk redudansi data, maka didapatkan rancangan akhir basis data yang akan digunakan untuk aplikasi ini. Berikut adalah struktur tabel hasil normalisasi yang dilakukan.

1. Rancangan Struktur Tabel

Setelah melakukan normalisasi tabel guna mengurangi atau menghilangkan kemungkinan terjadinya redudansi data, maka didapatkan rancangan basis data yang dianggap normal oleh penulis. Entitas-entitas yang sudah ditentukan kemudian dikonversikan ke dalam bentuk tabel di dalam basis data. Sedangkan atribut-atribut dari entitas tersebut dikonversikan ke dalam bentuk kolom-kolom pada tabel. Kolom-kolom tersebut akan ditentukan jenis tipe datanya dan panjang dari data yang dapat disimpan di dalamnya. Berikut adalah struktur tabel dari hasil konversi dan hasil normalisasi:

a. Tabel Personal

Berguna untuk menyimpan data-data personal seseorang seperti nama lengkap, nama alias, tanggal lahir dan lain-lain. Berikut adalah struktur tabel dari tabel personal yang ditunjukkan pada tabel 3.4.

Tabel 3.4 Struktur Tabel Personal

FIELD	TYPE	SIZE	KETERANGAN
id	Integer	3	<i>Primary Key</i> (id personal)
nama	Varchar	50	Nama Lengkap

nama_alias	Varchar	50	
tgl_lahir	Varchar	2	
bulan	Varchar	20	
tahun	Varchar	4	
jenis_kelamin	Varchar	15	
alamat	Varchar	300	
pekerjaan	Varchar	50	
nikah	Varchar	15	Status Pernikahan
no_tlp	Varchar	20	
status	Varchar	50	Meninggal atau Masih Hidup
tgl_m	Varchar	2	Tanggal meninggal
bulan_m	Varchar	20	Bulan meninggal
tahun_m	Varchar	4	Tahun meninggal
tempat_makam	Varchar	50	Tempat dimakamkan
foto	Varchar	50	

b. Tabel Keluarga

Berguna untuk menyimpan data-data keluarga seseorang seperti nama ayah, nama ibu dan jumlah anak. Berikut adalah struktur tabel dari tabel keluarga yang ditunjukkan pada tabel 3.5.

Tabel 3.5 Struktur Tabel Keluarga

FIELD	TYPE	SIZE	KETERANGAN
id_keluarga	Integer	4	<i>Primary Key</i>
id_ayah	Integer	3	<i>Foreign Key</i>
id_ibu	Integer	3	<i>Foreign Key</i>
jml_anak	Integer	3	Jumlah Anak

c. Tabel Ayah

Berguna untuk menyimpan data-data ayah. Berikut adalah struktur tabel dari tabel ayah yang ditunjukkan pada tabel 3.6.

Tabel 3.6 Struktur Tabel Ayah

FIELD	TYPE	SIZE	KETERANGAN
id_ayah	Integer	3	<i>Primary Key</i>
id	Integer	3	<i>Foreign Key</i> (Id personal)

d. Tabel Ibu

Berguna untuk menyimpan data-data ibu. Berikut adalah struktur tabel dari tabel ibu yang ditunjukkan pada tabel 3.7.

Tabel 3.7 Struktur Tabel Ibu

FIELD	TYPE	SIZE	KETERANGAN
id_ibu	Integer	3	<i>Primary Key</i>
id	Integer	3	<i>Foreign Key</i> (Id personal)

e. Tabel Anak

Berguna untuk menyimpan data-data anak. Berikut adalah struktur tabel dari tabel personal yang ditunjukkan pada tabel 3.8.

Tabel 3.8 Struktur Tabel Anak

FIELD	TYPE	SIZE	KETERANGAN
id_anak	Integer	3	<i>Primary Key</i>
id	Integer	3	<i>Foreign Key</i> (Id personal)
id_keluarga	Integer	4	<i>Foreign Key</i>

f. Tabel Admin

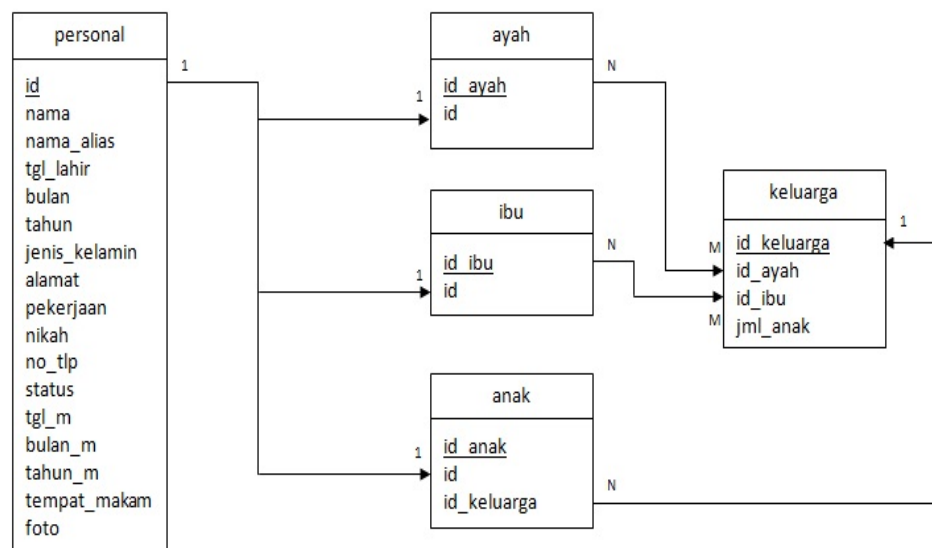
Berguna untuk menyimpan data-data personal seseorang seperti nama lengkap, nama alias, tanggal lahir dan lain-lain. Berikut adalah struktur tabel dari tabel personal yang ditunjukkan pada tabel 3.9.

Tabel 3.9 Struktur Tabel Admin

FIELD	TYPE	SIZE	KETERANGAN
id_admin	Integer	3	<i>Primary Key</i>
uname	varchar	20	
paswd	Integer	100	Password

2. Entity Relationship Diagram(ERD)

Basisdata relasioanal tidak akan pernah lepas dengan yang namanya *Entity Relationship Diagram*. Diagram ini akan memodelkan relasi atau hubungan antara *entity* (entitas). Entitas ini merupakan individu yang mewakili sesuatu yang nyata dan dapat dibedakan dari satu dengan yang lain. Setiap entitas akan memiliki atribut-atribut yang melekat pada dirinya. Dengan menggunakan *Entity Relationship Diagram* ini, dapat diketahui hubungan antar entitas. Dengan adanya hubungan antar entitas ini, dalam pengaplikasiannya, dalam hal ini adalah silsilah keluarga, akan menjadi lebih mudah dalam mencari setiap keturunan maupun setiap moyang dari seseorang. Berikut gambar ERD dari aplikasi ini yang ditunjukkan pada gambar 3.5.

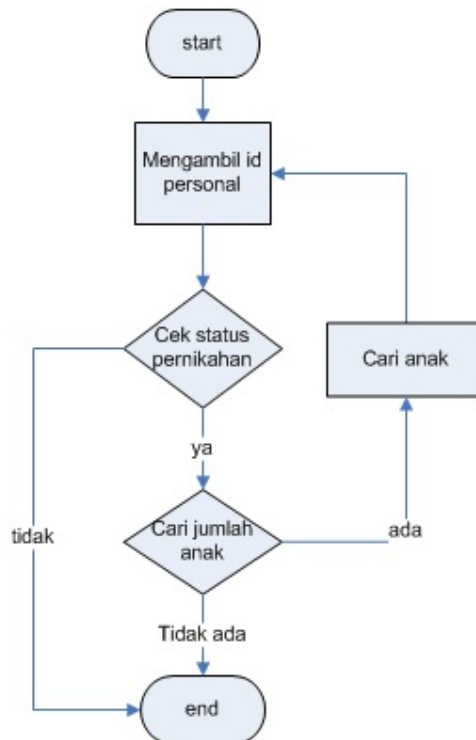


Gambar 3.5 Entity Relationship Diagram(ERD)

3.2.2.3 Flowchart

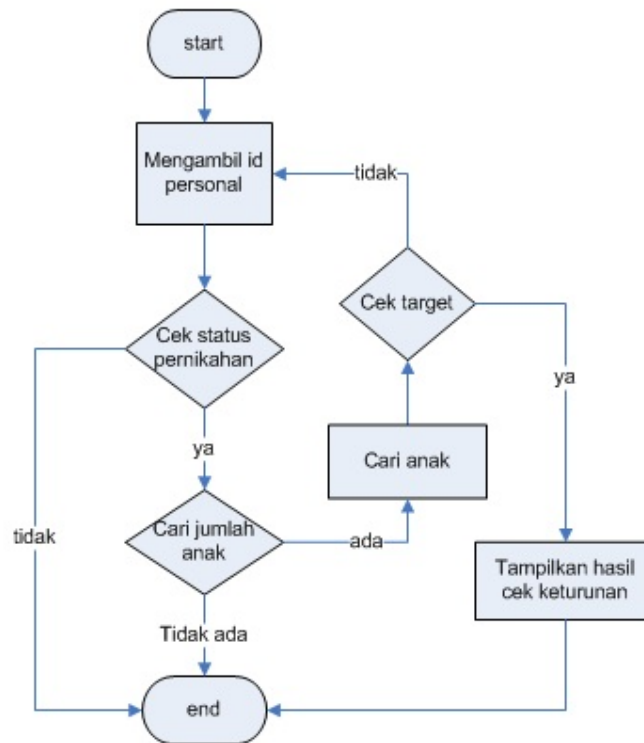
Pada aplikasi ini terdapat fitur pencarian dan pengecekan baik itu keturunan maupun moyang. Untuk pencarian keturunan mengadopsi algoritma pencarian pada struktur data berbentuk tree yaitu *depth-first*

dimana akan melakukan pencarian pada satu node terlebih dahulu sampai dengan level terbawah pada node tersebut, kemudian akan melakukan *backtracking* atau melangkah balik ke node berikutnya dan melakukan hal yang sama sampai dengan semua node dilewati . Berikut adalah *flowchart* untuk proses pencarian keturunan.



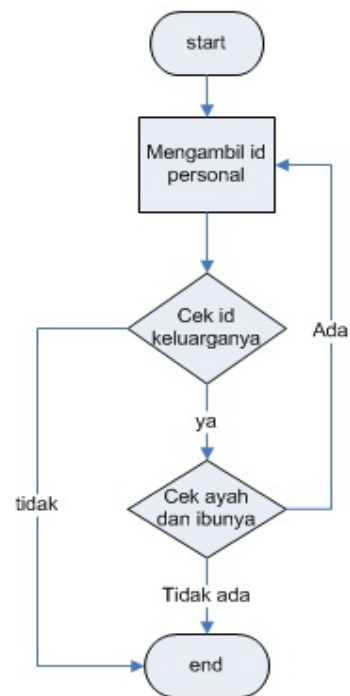
Gambar 3.6 Flowchart Proses Pencarian Keturunan

Selain hanya mencari keturunan, terdapat juga fitur untuk mengecek keturunan dari seseorang. Hal yang sama akan dilakukan dengan proses pencarian keturunan seperti di atas, namun terdapat sedikit penambahan proses yakni pengecekan dengan target. Berikut adalah gambar dari *flowchart* untuk proses pengecekan keturunan seseorang.

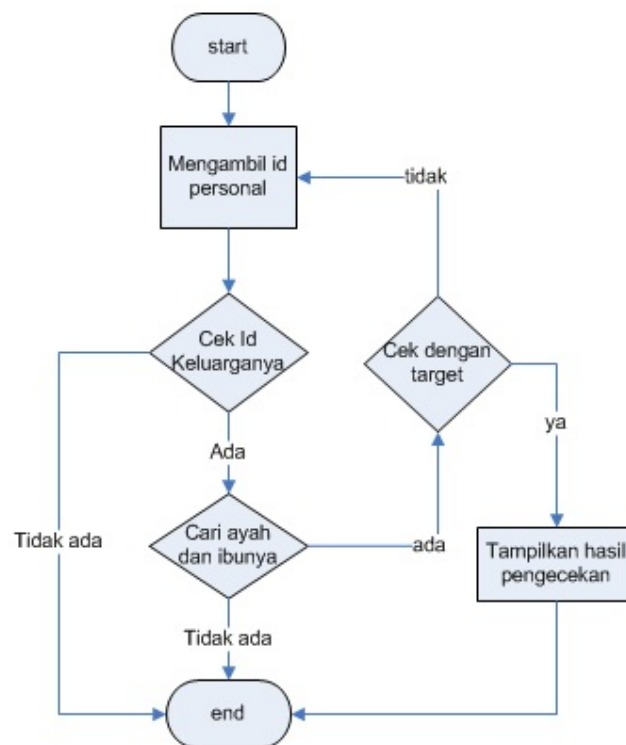


Gambar 3.7 Flowchart Pengecekan Keturunan

Selain pencarian keturunan, terdapat juga pencarian untuk moyang. Pada proses pencarian moyang ini mengadopsi pencarian pada struktur data berbentuk tree yaitu *depth-first*, dimana akan melakukan pencarian pada satu node terlebih dahulu sampai dengan level terbawah pada node tersebut, kemudian akan melakukan *backtracking* atau melangkah balik ke node berikutnya dan melakukan hal yang sama sampai dengan semua node dilewati. Hal yang dilakukan hampir sama, hanya saja terdapat proses tambahan yaitu proses pengecekan terhadap target. Berikut adalah *flow-chart* untuk kedua proses tersebut.



Gambar 3.8 Flowchart Proses Pencarian Moyang



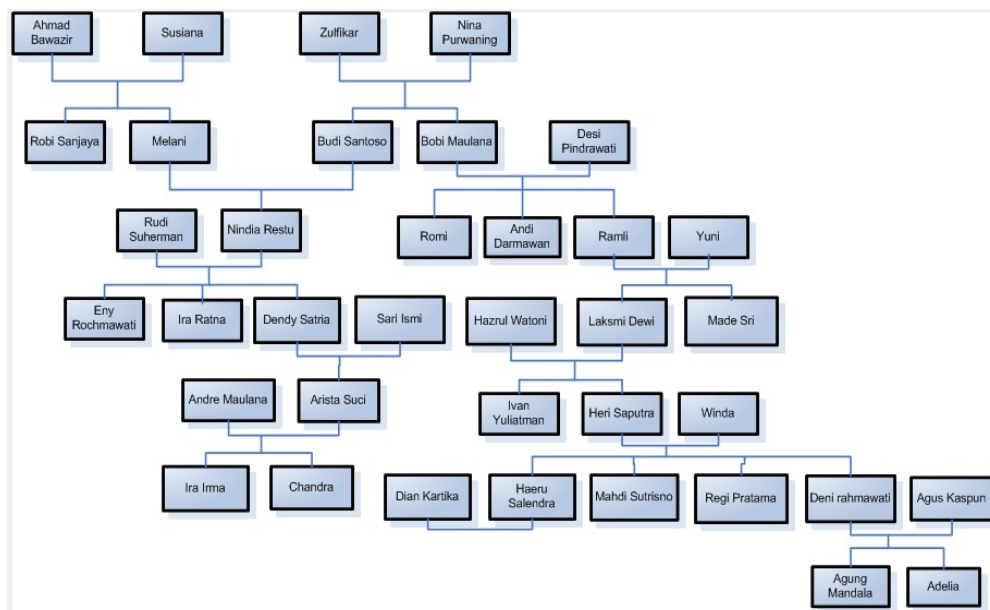
Gambar 3.9 Flowchart Proses Pengecekan Moyang

3.2.3 Implementasi sistem.

Setelah desain yang diinginkan didapatkan, kemudian dibentuk ke dalam model pemograman untuk mendapatkan hasil yang sesuai. Peneliti dalam hal ini menggunakan bahasa pemograman PHP (*PHP Hypertext Preprocessor*) yang merupakan bahasa pemograman berbasis web yang memiliki kemampuan memproses data secara dinamis, dan juga menggunakan MySQL sebagai basis data yang digunakan untuk menyimpan semua data yang diperlukan dan dapat digunakan kembali secara cepat dan mudah. Selain PHP dan MySQL, peneliti juga menggunakan Notepad++ sebagai editor teks untuk menuliskan baris-baris kode dan Wamp Server sebagai web server lokal.

3.3 Skenario Pengujian Sistem

Setelah implementasi selesai dan dianggap sempurna, maka aplikasi ini akan diuji sesuai dengan fungsi dan tujuannya untuk mencari kekurangan-kekurangan yang ada pada sistem yang nantinya dapat diperbaharui. Untuk melakukan pengujian ini, diperlukan sebuah data. Berikut adalah data uji yang digunakan dalam bentuk sebuah pohon keluarga yang dibuat secara sembarang.



Gambar 3.10 Data Pohon Keluarga Sebagai Data Uji Sistem

Berikut adalah skenario pengujian sistem ini dapat dilihat pada tabel berikut.

Tabel 3.10 Skenario Pengujian Sistem

No	Deskripsi Pengujian	Skenario Pengujian	Hasil Yang Diharapkan
1	Menguji Fungsi <i>login</i>	Memasukkan data nama pengguna dan data kata sandi yang sesuai dengan apa yang dibuat oleh pemogram dan yang tidak sesuai	Jika sesuai dengan data yang sudah dibuat oleh pemogram, maka akan berhasil masuk ke sistem, jika tidak akan menampilkan pesan bahwa data <i>login</i> salah
2	Menguji fungsi manajemen data	Memasukkan data-data baru, seperti data personal, data keluarga, data ayah, data ibu dan data anak, kemudian mengubah data yang sudah ada, kemudian menghapus data.	Fungsi dari tambah data, ubah data, hapus data, bila dijalankan akan memberikan keluaran yang sesuai dengan apa yang diharapkan.
3	Menguji fungsi dari Cari Data	Memasukkan kata kunci pencarian pada kolom yang telah disediakan dengan kata kunci yang sesuai dengan yang ada dibasisdata dan yang tidak	Jika di dalam basisdata terdapat data yang sesuai dengan kata kunci yang dimasukkan pada kolom yang telah disediakan maka akan menampilkan semua yang sesuai dengan apa yang telah dimasukkan
4	Menguji fungsi	Mencoba mencari keturunan seseorang	Akan menampilkan seluruh keturunan yang

	pencarian keturunan		segaris dengan data yang menjadi acuan pencarian
5	Menguji fungsi pencarian moyang	Mencoba mencari moyang dari seseorang	Akan menampilkan seluruh moyang dari seseorang baik itu dari jalur ayah maupun jalur ibu
6	Menguji fungsi cek keturunan	Mengecek apakah benar keturunannya dengan menggunakan data yang benar-benar keturunannya dan menggunakan data yang bukan keturunannya	Jika memang benar keturunannya maka akan menampilkan pemberitahuan bahwa benar keturunannya, dan jika tidak akan menampilkan seluruh keturunan dari data acuan
7	Menguji fungsi cek moyang	Mengecek apakah benar moyangnya dengan menggunakan data yang benar-benar moyangnya dan menggunakan data yang bukan moyangnya	Jika memang benar moyangnya maka akan menampilkan pemberitahuan bahwa benar moyangnya, dan jika tidak akan menampilkan seluruh moyang dari data acuan

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi

4.1.1 Halaman Muka

Halaman muka pada aplikasi ini terbagi ke dalam dua bagian. Bagian pertama adalah halaman muka untuk pengguna biasa dan yang kedua adalah halaman muka untuk admin. Gambar 4.1 ini merupakan tampilan halaman muka untuk pengguna biasa dan pada gambar 4.2 merupakan tampilan halaman muka untuk admin.



Gambar 4.1 Tampilan Halaman Muka Untuk Pengguna Biasa



Gambar 4.2 Tampilan Halaman Muka Untuk Admin

Terdapat sedikit perbedaan antara tampilan halaman muka untuk pengguna biasa dan untuk admin. Perbedaannya terdapat pada bagian menu aplikasi. Untuk pengguna biasa tidak ada menu untuk manajemen data, seperti : tambah data, ubah data dan hapus data.

4.1.2 Halaman *Login*

Setelah proses desain antarmuka untuk halaman *login*, gambar 4.3 berikut adalah hasil dari desain tersebut.

Gambar 4.3 Tampilan Halaman *Login* Admin

4.1.3 Halaman Cari Data

Halaman cari data ini diperutukkan bagi semua pengguna, baik itu admin maupun pengguna biasa. Gambar 4.4 berikut adalah implementasi untuk halaman cari data.

NO	NAMA LENGKAP	NAMA ALIAS	DETAIL		
1	Adelia		INDIVIDU	KETURUNAN	MOYANG
2	Agung Mandala		INDIVIDU	KETURUNAN	MOYANG
3	Agus Kaspun	Gotes	INDIVIDU	KETURUNAN	MOYANG
4	Ahmad Bawazir	Azir	INDIVIDU	KETURUNAN	MOYANG
5	Andi Darmawan		INDIVIDU	KETURUNAN	MOYANG
6	Andre Maulana		INDIVIDU	KETURUNAN	MOYANG
7	Arista Suci		INDIVIDU	KETURUNAN	MOYANG
8	Bobi Maulana		INDIVIDU	KETURUNAN	MOYANG
9	Budi Santoso		INDIVIDU	KETURUNAN	MOYANG
10	Chandra		INDIVIDU	KETURUNAN	MOYANG

Pages: 1/4 total Record(s): 37

Aplikasi Silsilah Keluarga, ©

Gambar 4.4 Tampilan Halaman Cari Data

4.1.4 Halaman Cek Keturunan

Sama halnya dengan halaman cari data, halaman cek keturunan juga diperutukkan bagi semua pengguna. Gambar 4.5 berikut adalah implementasi untuk halaman cek keturunan.

CEK KETURUNAN	
Nama Moyang	Adelia
Nama Keturunan	Adelia
cek reset batal	

Aplikasi Silsilah Keluarga, ©

Gambar 4.5 Tampilan Halaman Cek Keturunan

4.2 Pembahasan

4.2.1 Pengelolaan Data

Pada halaman muka untuk admin, terdapat *hyperlink* "Tambah Data". Pada *hyperlink* tersebut terdapat *dropdown* menu yang isinya untuk menambah data. Data yang bisa ditambahkan antara lain adalah data personal, data keluarga, data ayah, data ibu dan data anak.

a. Data Personal

Jika memilih tambah data untuk data personal, maka aplikasi akan menampilkan formulir untuk menambahkan data personal yang ditunjukkan pada gambar 4.6. Formulir ini digunakan untuk mengisi data personal yang dibutuhkan antara lain adalah id personal, nama lengkap, nama alias, tanggal lahir, jenis kelamin, alamat, pekerjaan, status pernikahan, nomor telepon, status kehidupan(meninggal dunia atau tidak), tempat makam dan foto.

TAMBAH DATA PERSONAL		
Id Personal	38	
Nama Lengkap	Nama Lengkap *	
Nama Alias	Nama Alias	
Tanggal Lahir	Tidak Tahu ▼ Tidak Tahu ▼ Tidak Tahu ▼	
Jenis Kelamin	☐ Laki_laki ☐ Perempuan	
Alamat		
Pekerjaan	Pekerjaan	
Status Pernikahan	☐ Belum Menikah ☐ Menikah	
No. Tlp	No. Telepon	
Status Kehidupan	☐ Meninggal ☐ Masih Hidup	
Tanggal Meninggal	Tidak Tahu ▼ Tidak Tahu ▼ Tidak Tahu ▼ **	
Tempat Makam		
Foto	Pilih File Tidak ada file yang dipilih	
tambah reset batal		
*harus diisi **diisi jika meninggal		

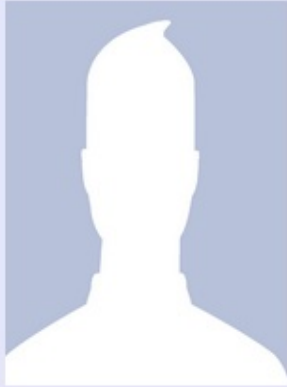
Gambar 4.6 Formulir Tambah Data Personal

Setelah data personal ini terisi kemudian berhasil ditambahkan ke dalam *database*, maka akan tampil di halaman personal seperti pada gambar 4.7 berikut ini.

DATA PERSONAL				
NO	NAMA LENGKAP	AKSI		
1	Adelia	Edit	Hapus	Detail
2	Agung Mandala	Edit	Hapus	Detail
3	Agus Kaspun	Edit	Hapus	Detail
4	Ahmad Bawazir	Edit	Hapus	Detail
5	Andi Darmawan	Edit	Hapus	Detail
6	Andre Maulana	Edit	Hapus	Detail
7	Arista Suci	Edit	Hapus	Detail
8	Bobo Maulana	Edit	Hapus	Detail
9	Budi Santoso	Edit	Hapus	Detail
10	Chandra	Edit	Hapus	Detail
		NEXT		
Pages: 1/4 total Record(s): 37				

Gambar 4.7 Tampilan Halaman Personal

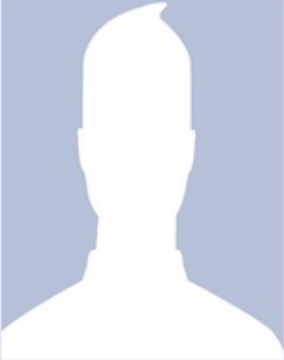
Pada halaman ini terdapat beberapa menu yang dapat dilakukan terhadap data personal, antara lain adalah edit, hapus dan detail. Edit sendiri berfungsi untuk mengubah data yang sudah ada menjadi jadi baru yang lebih akurat. Kemudian hapus, berfungsi untuk menghapus data yang sudah ada dan tidak akan ditampilkan lagi di layar. Menu lainnya adalah menu detail, ini adalah menu untuk melihat detail dari data personal yang telah diisi. Berikut adalah tampilan dari halaman detail personal.

Home Data Tambah Data Cari Data Logout	
DETAIL PERSONAL	
	
Nama Lengkap	Dendy Satria
Nama Alias	
Tanggal Lahir	16-Maret-1990
Jenis Kelamin	Laki-Laki
Alamat	Merca
Pekerjaan	Perawat
Status Pernikahan	Menikah
No. Tlp	
Aksi	Keluarga Edit Hapus Kembali

Gambar 4.8 Tampilan Halaman Detail

Pada halaman detail ini, terdapat dua jenis tampilan. Yang pertama adalah seperti yang tampak pada gambar 4.8 di mana seseorang yang masih hidup akan memiliki tampilan detail personal seperti demikian. Bagi seseorang yang

sudah meninggal dunia, maka akan tampil informasi tambahan seperti tanggal meninggal dan tempat dimakamkan seperti tampak pada gambar 4.9 berikut ini.

Home Data Tambah Data Cari Data Logout	
DETAIL PERSONAL	
	
Nama Lengkap	Ahmad Bawazir
Nama Alias	Azir
Tanggal Lahir	00-0000-0000
Jenis Kelamin	Laki-Laki
Alamat	
Pekerjaan	
Status Pernikahan	Menikah
No. Tlp	
Meninggal/Hidup	Meninggal
Tanggal Meninggal	00-0000-0000
Tempat Makam	
Aksi	Keluarga Edit Hapus Kembali

Gambar 4.9 Tampilan Halaman Detail Yang Sudah Meninggal

Setelah memilih detail personal, di sana juga terdapat beberapa pilihan antara lain adalah keluarga, edit, hapus dan kembali. Jika memilih keluarga, maka akan menampilkan informasi keluarga dari individu tersebut. Jika individu tersebut sudah menikah, maka akan terdapat informasi keluarga dari

individu tersebut. Namun jika belum menikah, tidak ada informasi tentang keluarga dari individu tersebut. Kemudian pilihan edit, adalah untuk merubah isi dari data individu tersebut yang tampilannya seperti pada gambar 4.10 berikut.

EDIT DATA PERSONAL	
Nama Lengkap	Dendy Satria
Nama Alias	Nama Alias
Tanggal Lahir	16 Maret 1990
Jenis Kelamin	☒ Laki_laki ☐ Perempuan
Alamat	Merca
Pekerjaan	Perawat
Status Kehidupan	☐ Belum Menikah ☒ Menikah
No. Tlp	No. Telepon
Status	☐ Meninggal ☒ Masih Hidup
Tanggal Meninggal	Tidak Tahu Tidak Tahu Tidak Tahu
Tempat Makam	
Foto	Pilih File Tidak ada file yang dipilih **
edit batal	
<i>**kosongkan jika menggunakan gambar yang lama</i>	

Gambar 4.10 Tampilan Halaman Edit Personal

Kemudian menu hapus, digunakan untuk menghapus data yang sudah ada dari *database*, sehingga tidak akan ditampilkan lagi dilayar.

b. Data Keluarga

Formulir untuk menambahkan data keluarga akan ditampilkan seperti pada gambar 4.11. Pada saat mengisi data keluarga ini hanya dalam ruang lingkup keluarga kecil yang hanya terdiri dari Ayah, Ibu dan Anak. Pada formulir ini akan digunakan untuk mengisi data yang dibutuhkan untuk sebuah keluarga seperti id keluarga, ayah, ibu dan jumlah anak yang dimiliki.

TAMBAH DATA KELUARGA	
Id Keluarga	14
Nama Ayah	Nama Ayah
Nama Ibu	Nama Ibu
Jumlah Anak	0
lanjut reset batal	

Gambar 4.11 Formulir Tambah Data Keluarga

Setelah mengisi nama ayah, nama ibu dan jumlah anak, maka akan dilanjutkan ke tahap selanjutnya yaitu menambahkan nama anak untuk keluarga tersebut, seperti yang tampak pada gambar 4.12 berikut ini.

TAMBAH DATA ANAK	
Id Keluarga	14
Nama Ayah	Ahmad Bawazir
Nama Ibu	Susiana
Nama Anak 1	Nama Anak
Nama Anak 2	Nama Anak
tambah reset	

Gambar 4.12 Formulir Tambah Keluarga Lanjutan

Setelah selesai, maka akan ditampilkan halaman keluarga di mana akan menampilkan semua keluarga yang sudah ditambahkan ke dalam *database*. Gambar 4.13 berikut adalah tampilan dari halaman keluarga.

DATA KELUARGA						
NO	NAMA AYAH	NAMA IBU	JUMLAH ANAK	AKSI		
1	Rudi Suherman	Nindia Restu	3	Edit	Hapus	Detail
2	Ahmad Bawazir	Susiana	2	Edit	Hapus	Detail
3	Zulfikar	Nina Purwaning	2	Edit	Hapus	Detail
4	Budi Santoso	Melani	1	Edit	Hapus	Detail
5	Bobo Maulana	Desi Pindrawati	3	Edit	Hapus	Detail
6	Ramli	Yuni	2	Edit	Hapus	Detail
7	Dendy Satria	Sari Ismi	1	Edit	Hapus	Detail
8	Andre Maulana	Arista Suci	2	Edit	Hapus	Detail
9	Heri Saputra	Winda	4	Edit	Hapus	Detail
10	Haeru Salendra	Dian Kartika	0	Edit	Hapus	Detail
				NEXT		
				Pages: 1/2 total Record(s): 12		

Gambar 4.13 Tampilan Halaman Keluarga

Pada halaman ini akan menampilkan semua keluarga yang sudah tercatat di dalam *database*. Di sini terdapat informasi singkat dari keluarga seseorang antara lain adalah nama ayah, nama ibu dan jumlah anak. Kemudian di kolom aksi terdapat beberapa pilihan, yakni edit, hapus dan detail. Edit sendiri akan menampilkan formulir ubah data keluarga seperti yang tampak pada gambar 4.14 berikut.

EDIT DATA KELUARGA	
Id Keluarga	1
Nama Ayah	Ahmad Bawazir
Nama Ibu	Susiana
Jumlah Anak	2
Simpan	

Gambar 4.14 Tampilan Halaman Edit Keluarga

Di sini terdapat pilihan untuk mengubah ayah, ibu dan jumlah anak dari data keluarga sebelumnya. Kemudian menu hapus, menu ini digunakan untuk

menghapus data keluarga yang sudah ada. Dan yang terakhir adalah menu detail. Di sini akan menampilkan informasi keluarga itu secara menyeluruh, dalam artian akan menampilkan nama ayah, nama ibu, jumlah anak dan nama dari anak-anaknya. Gambar 4.15 berikut adalah tampilan dari halaman detail keluarga.

DETAIL DATA KELUARGA				
ID KELUARGA	NAMA AYAH	NAMA IBU	JUMLAH ANAK	AKSI
1	Ahmad Bawazir	Susiana	2	Aksi
NAMA ANAK	Melani			
NAMA ANAK	Robi Sanjaya			

Gambar 4.15 Tampilan Halaman Detail Keluarga

Pada halaman tersebut juga terdapat kolom aksi yang akan menampilkan *dropdown* menu. Isi dari aksi tersebut adalah edit dan hapus.

c. Data Ayah

Formulir untuk menambah data ayah akan ditampilkan seperti pada gambar 4.16. Untuk menambah data ayah, nantinya akan menampilkan semua yang ada di data personal yang sudah menikah dan berjenis kelamin laki-laki. Sehingga pada saat memilih nama akan menjadi lebih mudah.

TAMBAH DATA AYAH	
Id Ayah	14
Nama Ayah	Nama Ayah ▼
tambah reset batal	

Gambar 4.16 Formulir Tambah Data Ayah

c. Data Ibu

Pada gambar 4.17 menunjukkan formulir untuk tambah ibu. Pada dasarnya sama dengan formulir tambah data ayah, hanya saja akan menampilkan data dari data personal yang sudah menikah dan berjenis kelamin perempuan.

TAMBAH DATA IBU	
Id Ibu	15
Nama Ibu	Nama Ibu ▼
tambah reset batal	

Gambar 4.17 Formulir Tambah Data Ibu

d. Data Anak

Untuk formulir tambah data anak, ditunjukkan seperti pada gambar 4.18. Untuk data anak, akan menampilkan semua data yang ada di data personal, baik yang sudah menikah mau tidak serta yang berjenis kelamin laki-laki maupun perempuan.

TAMBAH DATA ANAK	
Nama Anak	Nama Anak ▼
tambah reset batal	

Gambar 4.18 Formulir Tambah Data Anak

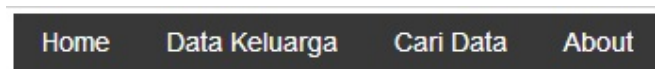
4.2.2 Pencarian Data

Pada menu Cari data terdapat tiga buah pilihan, yakni cari, cek keturunan dan cek moyang. Pada menu cari data ini, berfungsi untuk mencari data personal atau individu dengan memasukkan kata kunci atau dalam hal ini menggunakan nama untuk mencari data. Sedangkan untuk cek keturunan adalah untuk mengecek kebenaran keturunan dari sebuah silsilah keluarga, dan untuk cek moyang adalah untuk mengecek kebenaran moyang dari seseorang.

Dengan menggunakan rancangan basis data seperti tergambar pada ERD yang sudah dijelaskan di bab sebelumnya, sebuah silsilah keluarga pada saat memasukkan ke dalam *database*, akan menggunakan kelompok keluarga terkecil yang dalam artian hanya meliputi ayah, ibu dan anak. Dengan

menggunakan keluarga kecil ini, nantinya dapat mencari keturunan maupun moyang dari seseorang. Dikarenakan menggunakan basis data relasional, maka akan semakin mudah dalam hal pencariannya.

Proses pencarian ini dapat dilakukan oleh semua pengguna, baik itu pengguna biasa maupun admin dari aplikasi ini. Untuk melakukan pencarian keturunan ini, dapat dilakukan dengan meng-klik *hyperlink* "Cari Data" pada bagian menu yang terdapat di aplikasi.



Gambar 4.19 Tampilan Menu Untuk Pengguna Biasa



Gambar 4.20 Tampilan Menu Untuk Admin

Setelah meng-klik *hyperlink* "Cari Data" maka akan tampil halaman pencarian seperti pada gambar 4.21 berikut.

Masukkan Nama

cari !

NO	NAMA LENGKAP	NAMA ALIAS	DETAIL		
1	Adelia		INDIVIDU	KETURUNAN	MOYANG
2	Agung Mandala		INDIVIDU	KETURUNAN	MOYANG
3	Agus Kaspun	Gotes	INDIVIDU	KETURUNAN	MOYANG
4	Ahmad Bawazir	Azir	INDIVIDU	KETURUNAN	MOYANG
5	Andi Darmawan		INDIVIDU	KETURUNAN	MOYANG
6	Andre Maulana		INDIVIDU	KETURUNAN	MOYANG
7	Arista Suci		INDIVIDU	KETURUNAN	MOYANG
8	Bobbi Maulana		INDIVIDU	KETURUNAN	MOYANG
9	Budi Santoso		INDIVIDU	KETURUNAN	MOYANG
10	Chandra		INDIVIDU	KETURUNAN	MOYANG

NEXT

Pages: 1/4

total Record(s): 37

Gambar 4.21 Tampilan Halaman Cari Data

Dari halaman tersebut akan tampil data individu yang dilengkapi dengan *hyperlink* untuk detail, terdapat detail untuk individu, detail keturunan dan detail moyang. Ketika memilih detail keturunan, maka akan menampilkan seluruh keturunan dari individu yang dipilih. Berikut adalah contoh hasil dari pencarian keturunan salah satu dari data individu yang ada pada *database* seperti gambar 4.22 berikut ini.

KETURUNAN DARI 'Bobi Maulana'	
•	Bobi Maulana
•	Andi Darmawan
•	Romi
•	Ramli
•	Laksmi Dewi
•	Ivan Yuliatman
•	Heri Saputra
•	Haeru Salendra
•	Mahdi Sutrisno
•	Regi Pratama
•	Deni Rahmawati
•	Agung Mandala
•	Adelia
•	Made Sri

Gambar 4.22 Hasil Dari Pencarian Keturunan

Proses pencarian itu sendiri menggunakan konsep *depth-first*, dimana akan mencari pada satu cabang sampai dengan tidak ada lagi data yang ada pada cabang tersebut, kemudian akan *backtracking* atau melangkah balik ke cabang berikutnya dan melakukan hal yang sama sampai dengan semua cabang dilewati.

Selain proses pencarian keturunan ini, ada juga proses pencarian yang serupa namun terdapat proses pengecekan terhadap data yang dilewati pada saat proses pencarian. Untuk menggunakan fasilitas ini terdapat pada bagian *dropdown* "Cari Data" kemudian pilih "Cek Keturunan" seperti pada gambar 4.23 berikut ini.



Gambar 4.23 Menu *Dropdown* Cek Keturunan

Setelah menu "Cek Keturunan" dipilih, maka akan tampil formulir untuk menentukan antara individu siapa dengan siapa yang akan dilakukan pengecekan keturunan.

CEK KETURUNAN	
Nama Moyang	Bobi Maulana ▼
Nama Keturunan	Heri Saputra ▼
cek reset batal	

Gambar 4.24 Formulir Untuk Pengecekan Keturunan

Pada gambar di atas, akan dilakukan pengecekan apakah Heri Saputra adalah benar keturunan yang segaris dengan Bobi Maulana. Ketika tombol "cek" ditekan dan memang benar keturunan dari data yang dicari, maka akan menampilkan hasil seperti pada gambar 4.25 berikut.

HASIL CEK KETURUNAN
Heri Saputra adalah Memang Benar Keturunan dari Bobi Maulana

Gambar 4.25 Hasil Dari Pengecekan Keturunan

Namun jika tidak benar keturunan dari data yang dicari, maka akan menampilkan seluruh keturunan dari moyang, dalam hal ini adalah Bobi Maulana seperti pada gambar 4.22 di atas.

Pada proses pengecekan ini, pada dasarnya sama dengan proses pencarian keturunan seperti di atas. Namun ada penambahan proses yaitu pengecekan target keturunannya. Dalam proses ini juga, setiap individu yang

berhasil ditemukan dalam proses pencarian akan dilanjutkan ke proses pengecekan. Sehingga nantinya dapat memberikan keterangan seperti pada gambar 4.25 di atas.

Kedua proses tersebut, baik itu proses pencarian tanpa ada pengecekan dan proses pencarian dengan pengecekan menggunakan sistem rekursif. Di mana terdapat fungsi yang digunakan untuk melakukan pencarian tanpa pengecekan maupun pencarian dengan pengecekan, di dalamnya akan memanggil fungsi itu sendiri. Sehingga dapat meringkas *source code* untuk proses pencariannya.

Selain pencarian keturunan dan mengecek keturunan, terdapat juga pencarian moyang dan pengecekan moyang. Dengan menggunakan metode *Breadth-first Search*, proses pencarian dan pengecekan moyang akan dilakukan ke samping kemudian dilanjutkan ke atas. Berikut adalah hasil untuk proses pencarian moyang. Dalam kasus ini akan dicari moyang dari Heri Saputra, maka akan menghasilkan seperti gambar 4.26 berikut:

MOYANG DARI 'Heri Saputra'
• Hazrul Watoni • Laksmi Dewi • Ramli • Bobi Maulana • Zulfikar • Nina Purwaning • Desi Pindrawati • Yuni

Gambar 4.26 Hasil Dari Pencarian Moyang

Kemudian dilakukan pengecekan moyang terhadap Heri Saputra, dan jika sesuai dengan data pohon keluarga, maka akan menghasilkan seperti pada gambar 4.27 berikut:

HASIL CEK MOYANG
Zulfikar adalah Memang Benar Moyang dari Heri Saputra

Gambar 4.27 Hasil Dari Pengecekan Moyang

Ketika pengecekan moyang ini tidak menemukan hasil yang sesuai, maka akan menampilkan seluruh moyang seperti pada gambar 4.26 di atas.

Tabel 4.1 Hasil Pengujian Sistem

No	Deskripsi Pengujian	Skenario Pengujian	Hasil Yang Diharapkan	Hasil Pengujian
1	Menguji Fungsi <i>login</i>	Memasukkan data nama pengguna dan data kata sandi yang sesuai dengan apa yang dibuat oleh pemogram dan yang tidak sesuai	Jika sesuai dengan data yang sudah dibuat oleh pemogram, maka akan berhasil masuk ke sistem, jika tidak akan menampilkan pesan bahwa data <i>login</i> salah	Berhasil masuk ke dalam sistem menggunakan data yang sesuai dengan nama pengguna dan kata sandi yang benar.
2	Menguji fungsi manajemen data	Memasukkan data-data baru, seperti data personal, data keluarga, data ayah, data ibu dan data anak, kemudian mengubah data yang sudah ada, kemudian menghapus data.	Fungsi dari tambah data, ubah data, hapus data, bila dijalankan akan memberikan keluaran yang sesuai dengan apa yang diharapkan.	Data berhasil ditambah, diubah, dihapus dari sistem.
3	Menguji fungsi dari	Memasukkan kata kunci	Jika di dalam basisdata terdapat	Berhasil menampilkan

	Cari Data	pencarian pada kolom yang telah disediakan dengan kata kunci yang sesuai dengan yang ada dibasisdata dan yang tidak	data yang sesuai dengan kata kunci yang dimasukkan pada kolom yang telah disediakan maka akan menampilkan semua yang sesuai dengan apa yang telah dimasukkan	seluruh data yang sesuai dengan apa yang menjadi kata kunci pencarian.
4	Menguji fungsi pencarian keturunan	Mencoba mencari keturunan seseorang	Akan menampilkan seluruh keturunan yang segaris dengan data yang menjadi acuan pencarian	Berhasil menampilkan seluruh keturunan dari data acuan
5	Menguji fungsi pencarian moyang	Mencoba mencari moyang dari seseorang	Akan menampilkan seluruh moyang dari seseorang baik itu dari jalur ayah maupun jalur ibu	Berhasil menampilkan moyang dari jalur ayah maupun jalur ibu
6	Menguji fungsi cek keturunan	Mengecek apakah benar keturunannya dengan menggunakan data yang benar-benar keturunannya dan	Jika memang benar keturunannya maka akan menampilkan pemberitahuan bahwa benar keturunannya, dan jika tidak akan menampilkan seluruh keturunan	Berhasil menampilkan pemberitahuan bahwa benar keturunannya dan menampilkan seluruh keturunan dari

		menggunakan data yang bukan keturunannya	dari data acuan	data acuan jika bukan keturunannya
7	Menguji fungsi cek moyang	Mengecek apakah benar moyangnya dengan menggunakan data yang benar-benar moyangnya dan menggunakan data yang bukan moyangnya	Jika memang benar moyangnya maka akan menampilkan pemberitahuan bahwa benar moyangnya, dan jika tidak akan menampilkan seluruh moyang dari data acuan	Berhasil menampilkan pemberitahuan bahwa benar moyangnya dan menampilkan seluruh moyang dari data acuan jika bukan keturunannya

4.3 Kelebihan dan Kekurangan Sistem

Dengan melakukan pengujian sistem, maka dapat memberikan gambaran kepada pemrogram seberapa baik kualitas dari sistem yang telah dikembangkan. Setiap program ataupun perangkat lunak yang dikembangkan pasti memiliki kelebihan dan kekurangan.

4.3.1 Kelebihan Sistem

Berikut adalah beberapa kelebihan dari sistem berdasarkan hasil pengujian di atas.

1. Dapat mencari seluruh keturunan dari seseorang tanpa terlewatkan satupun.
2. Dapat mencari seluruh moyang dari seseorang, baik itu dari jalur ayah maupun jalur ibunya.
3. Memiliki fitur pencarian

4.3.2 Kekurangan Sistem

Berikut adalah beberapa kekurangan dari sistem.

1. Tidak dapat menjelaskan keturunan dari generasi ke generasi.
2. Tidak dapat menjelaskan moyang dari generasi ke generasi.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Setelah dilakukan pengujian dengan menggunakan data yang ada, seperti yang sudah tertera di bab sebelumnya, di mana dilakukan pencarian keturunan dari Bobi Maulana, kemudian mengecek hubungan keturunan antara Bobi Maulana dengan Heri Saputra. Dengan melihat hasil yang telah dicapai, maka dapat diambil kesimpulan atas aplikasi pohon keluarga (silsilah) dengan menggunakan basisdata relasional selama dalam proses pengembangan aplikasi tersebut antara lain:

1. Semua fitur manajemen data berjalan dengan baik dan sesuai dengan fungsinya masing-masing, baik itu untuk menambah data, mengubah data dan menghapus data.
2. Fitur pencarian keturunan maupun moyang dapat berjalan sesuai dengan apa yang diharapkan oleh pemrogram. Sehingga pengguna dapat melihat siapa saja keturunan maupun moyang seseorang yang telah dipilih.
3. Fitur pengecekan keturunan juga berfungsi sebagaimana yang diharapkan oleh pemrogram. Dengan adanya fitur ini, pengguna dapat mengetahui kebenaran dari keturunan seseorang.
4. Aplikasi ini juga sudah dilengkapi dengan fasilitas pencarian, sehingga pengguna akan menjadi lebih mudah dalam mencari data yang diinginkan.

5.2 Saran

Sesuai dengan apa yang telah dikemukakan oleh penulis pada bab sebelumnya, setiap aplikasi maupun perangkat lunak yang dikembangkan tentu tidak luput dari kekurangan. Oleh karena itu perlu adanya pengembangan lebih lanjut lagi agar aplikasi ini menjadi sebuah aplikasi yang lebih sempurna. Berikut adalah beberapa saran untuk menyempurnakan aplikasi pohon keluarga (silsilah) dengan menggunakan basisdata relasional, antara lain:

1. Menambahkan penjelasan garis keturunan maupun moyang dari generasi ke generasi.
2. Belum dapat mendiskripsikan hubungan seseorang.
3. Memperbaiki tampilan agar lebih bagus.

DAFTAR PUSTAKA

Kadir, Abdul. 1998. Konsep & Tuntunan Praktis Basis Data : penerbit ANDI, Yogyakarta.

Marcuszakaria, Teddy & Agus Prijono. 2005. Konsep Dan Implementasi Struktur Data : penerbit Informatika.

Fathansyah. 1999. Buku Teks Ilmu Komputer Basis Data : penerbit Informatika, Bandung.

Nugroho, Bunafit. 2004. Cascading Style Sheet (CSS). penerbit : Gava Madia.

Simarmata, Janner & Iman Prayudi. 2006. Basis Data: penerbit ANDI, Yogyakarta.

Kadir, Abdul. 2008. Dasar Pemograman Web Menggunakan PHP. penerbit : Andi, Yogyakarta.

<http://justizzah.blogspot.com/2013/05/sistem-basis-data-integritas-data.html>. Diakses pada tanggal 8 Oktober 2013