BAB I

PENDAHULUAN

A. Latar Belakang Masalah

Kulit merupakan organ tubuh terbesar yang menutupi seluruh permukaannya. Kulit berfungsi sebagai pelindung tubuh dari cedera dan patogen. Kulit juga mengatur suhu tubuh, mengendalikan kehilangan cairan yang tak terasa (insensible fluid loss), serta menyimpan vitamin D, lemak, dan air.

Kulit juga terdiri dari tiga lapisan . Yang pertama merupakan epidermis yang adalah lapisan pelindung dan tanpa pembuluh darah dari kulit. Lapisan ini 90% terdiri dari sel skuamosa atau keratinosit, yang menghasilkan keratin. Fungsi utama keratinosit adalah melindungi tubuh dari kerusakan kulit akibat panas, kehilangan cairan, virus, jamur, bakteri, dan parasit. Setiap bagian tubuh memiliki ketebalan epidermis yang berbeda; paling tebal di telapak kaki dan tangan, sedangkan paling tipis di kelopak mata (sekitar 0.05 mm). Lapisan kudua adalah dermis yang terletak di antara epidermis dan hipodermis. Dermis terdiri dari dua lapisan, yaitu dermis papilla dan retikular, yang membantu melindungi tubuh dari tegangan dan tekanan. Dermis juga mengandung kelenjar apokrin, jaringan penghubung, pembuluh darah, folikel rambut, kelenjar keringat, dan pembuluh limfatik. Pada lapisan ini termoreseptor dapat ditemukan, yang mendeteksi panas dan mekanoreseptor, yang peka terhadap sentuhan. Lapisan ketiga merupakan hypodermis , Hipodermis merupakan lapisan kulit yang paling tebal. Lapisan ini dilekatkan oleh serat elastin dan kolagen pada dermis. Hipodermis sebagian besar terdiri dari adiposa atau sel yang mengumpulkan dan menyimpan lemak.

Hipodermis juga mengandung pembuluh darah dan saraf yang lebih besar daripada yang ditemukan di dermis.

Penyakit kulit merupakan salah satu penyakit yang banyak terjadi pada masyarakat sekarang seperti ruam yang merupakan istilah untuk semua jenis perubahan warna atau peradangan kulit yang membuat kulit terlihat tidak normal. Contohnya adalah cacar air, rubella, urtikaria, intertrigo (ruam kulit di lipatan tubuh), rosacea, dan eksim. Ruam dapat disebabkan oleh obat-obatan, suhu ekstrim, gigitan serangga, kondisi medis, atau infeksi. Jerawat yang merupakan gangguan kulit yang paling umum. Setidaknya 85% orang pasti pernah terkena jerawat, yang muncul saat kulit tersumbat oleh bakteri, kulit mati, atau minyak. Dan banyak lagi seperti kanker kulit, kurap, ketombe dan lain-lainnya.

Namum masyarakat masih memiliki pengetahuan minim tentang penyakit kulit dikarenakan banyaknya jenis penyakit kulit dan juga sumber yang tidak memuatkan semua jenis penyakit kulit. Di sini penulis terinspirasi untuk mengangkat judul "**Aplikasi Edukasi Penyakit Kulit Berbasis WEB**" yang mana pengetahuan tentang kulit dan penyakit kulit serta pengobatan terdapat pada sistem ini.

B. Rumusan Masalah

Berdasarkan latar belakang masalah yang sudah dilakukan maka rumusan masalah yang dikaji dalam penelitian ini adalah:

1. Bagaimana merancang sebuah aplikasi edukasi penyakit kulit berbasis *WEB*?
2. Bagaimana cara memasukkan informasi berkaitan penyakit kulit, pencegahan, gejala dan pengobatan pada aplikasi ?
3. Bagaimana user dapat melihat informasi berkaitan edukasi penyakit kulit ?

C. Batasan Masalah

Agar pembahasan ini lebih terarah dan menghindari pembahasan lebih luas. Maka pembahasan ini dibatasi dengan hal-hal berikut. Meliputi :

1. Aplikasi dibuat menggunakan bahasa program *Javascript, PHP, HTML* dan *CSS.*
2. Basis Data yang digunakan merupakan *MySQL*.
3. Aplikasi hanya meliputi penyakit kulit, gejala, pengcegahan dan pengobatan.

D. Tujuan Penelitian

1. Merancang sebuah aplikasi edukasi penyakit kulit berbasis *WEB.*
2. Admin dapat memasukkan informasi berkaitan penyakit kulit, pencegahan, gejala dan pengobatan pada sistem
3. User dapat melihat informasi berkaitan edukasi penyakit kulit .

E. Manfaat Penelitian

1. Bermanfaat untuk masyarakat dalam mengetahui penyakit kulit, pencegahan, gejala dan pengotannya

F. Sistematika Penulisan

Struktur penulisan tugas akhir ini adalah :

BAB I : PENDAHULUAN

Berisi tentang latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, metodologi penelitian, dan sistematika penulisan dari pembuatan tugas akhir ini.

BAB II : TINJAUAN PUSTAKA

Berisi tentang penelitian terdahulu dan dasar teori yang mendukung pembuatan tugas akhir ini.

BAB III : METODE PENELITIAN

Berisi tentang letak lokasi serta waktu, jenis penelitian, metode pengumpulan data, alat dan bahan penelitian, tahap penelitian, metode pengujian, dan anlisa sistem.

BAB IV : HASIL DAN PERANCANGAN SISTEM

Pada Bab ini akan dibahas bagaimana cara menghubungkan komponen satu dengan yang lain setelah itu dilakukan pengujian pada masing-masing komponen.

BAB V : PENUTUP

Pada Bab ini berisikan kesimpulan dan saran.

DAFTAR PUSTAKA

LAMPIRAN

BAB II

TINJAUAN PUSTAKA

A. Penelitian Terdahulu

1. "Sistem Pakar Diagnosa Penyakit Alergi Pada Anak Berbasis WEB" (Jarti & Trisno, 2021). Kesimpulan dari penelitian ini adalah.
   a) Aplikasi sistem pakar yang dirancang mampu mendiagnosa penyakit pada anak berdasarkan gejala-gejala yang diinput oleh user.
   b) Aplikasi sistem pakar mempermudahkan user mendapatkan untuk mendapatkan data-data atau informasi yang diperlukan oleh user tentang penyakit alergi pada anak.

   Persamaan penelitian diatas dengan penulis adalah topik yang diteliti yang mana merupakan penyakit yang berlaku pada masyarakat umum.

   Perbedaan penelitian diatas dengan penulis adalah diatas meneliti penyakit alergi pada anak manakala penulis meneliti penyakit kulit.
2. "Klasifikasi Penyakit Kulit Menggunakan Algoritma *Naïve Bayes* Berdasarkan Tekstur Warna Berbasis *Android*" (Perangin-angin & Segala, 2021). Kesimpulan dari penelitian adalah:

   Berdasarkan penelitian Skripsi di atas bisa di sampaikan bahwa penerapan algoritma naive bayes untuk deteksi penyakit kulit berdasarkan tekstur warna dapat mengiplementasikan teknis klasifikasi dan pemrosesan citra dalam membantu proses indentifikasi penyakit kulit. Dari hasil penelitian bahwa klasifikasi penyakit kulit ini (eksim, jerawat, cacar air, campak, dan psioriasis). Dapat mengindentifikasi melalui algoritma naïve bayes dan

memperoleh akurasi sebesar 75%, dan berjalan dengan baik. Sistem dapat mempercepat proses klasifikasi penyakit kulit dengan cepat dan akurat

Persamaan penelitian diatas dengan penulis adalah topik yang diteliti yang mana merupakan penyakit kulityang berlaku pada masyarakat umum.

Perbedaan penelitian diatas dengan penulis adalah penulis terarah kepada aplikasi manakala penelitian diatas menggunakan metode *naïve bias* untuk mendeteksi penyakit kulit.

B. Kulit

Kulit merupakan organ tubuh terbesar yang menutupi seluruh permukaannya. Kulit berfungsi sebagai pelindung tubuh dari cedera dan patogen. Kulit juga mengatur suhu tubuh, mengendalikan kehilangan cairan yang tak terasa (insensible fluid loss), serta menyimpan vitamin D, lemak, dan air. Kulit juga terdiri dari tiga lapisan . Yang pertama merupakan epidermis yang adalah lapisan pelindung dan tanpa pembuluh darah dari kulit. Lapisan ini 90% terdiri dari sel skuamosa atau keratinosit, yang menghasilkan keratin. Fungsi utama keratinosit adalah melindungi tubuh dari kerusakan kulit akibat panas, kehilangan cairan, virus, jamur, bakteri, dan parasit. Setiap bagian tubuh memiliki ketebalan epidermis yang berbeda; paling tebal di telapak kaki dan tangan, sedangkan paling tipis di kelopak mata (sekitar 0.05 mm) (Kusantati, 2008). Lapisan kedua adalah dermis yang terletak di antara epidermis dan hipodermis. Dermis terdiri dari dua lapisan, yaitu dermis papilla dan retikular, yang membantu melindungi tubuh dari tegangan dan tekanan. Dermis juga mengandung kelenjar apokrin, jaringan penghubung, pembuluh darah, folikel rambut, kelenjar keringat, dan pembuluh limfatik. Pada lapisan ini termoreseptor dapat ditemukan, yang mendeteksi panas

dan mekanoreseptor, yang peka terhadap sentuhan. Lapisan ketiga merupakan hypodermis , Hipodermis merupakan lapisan kulit yang paling tebal. Lapisan ini dilekatkan oleh serat elastin dan kolagen pada dermis. Hipodermis sebagian besar terdiri dari adiposa atau sel yang mengumpulkan dan menyimpan lemak. Hipodermis juga mengandung pembuluh darah dan saraf yang lebih besar daripada yang ditemukan di dermis.

C. Penyakit Kulit

Penyakit kulit merupakan salah satu penyakit yang banyak terjadi pada masyarakat sekarang seperti ruam yang merupakan istilah untuk semua jenis perubahan warna atau peradangan kulit yang membuat kulit terlihat tidak normal. Contohnya adalah cacar air, rubella, urtikaria, intertrigo (ruam kulit di lipatan tubuh), rosacea, dan eksim. Ruam dapat disebabkan oleh obat-obatan, suhu ekstrim, gigitan serangga, kondisi medis, atau infeksi. Jerawat yang merupakan gangguan kulit yang paling umum. Dilansir dari American Academy of Dermatology, sekitar 85% orang antara usia 12 - 24 mengalami setidaknya jerawat ringan dalam hidupnya. Gangguan kulit ini terjadi ketika folikel rambut tersumbat oleh minyak dan sel kulit mati. Hal tersebut menyebabkan peradangan serta penyumbatan pada pori-pori kulit. Peradangan ini ditandai dengan munculnya benjolan kecil yang terkadang berisi nanah di atas kulit.

D. Javascript

Javascript merupakan salah satu bahasa pemrograman web yang bisa membuat halaman *website* anda lebih menarik serta lebih hidup. Bahasa pemrograman *javascript* itu dijalankan di perangkat pengunjung website atau situs, bukan di server jadi berbeda dengan *PHP*.*Javascript* juga dikatakan berbeda dari *HTML* dan *CSS*. *HTML* itu fungsinya untuk mengatur tampilan konten dan *CSS* mengelola *layoutnya* jadi ketiganya sangatlah berbeda (Enterprise, 2019). Banyak yang mengibaratkan perbandingan *HTML*, *CSS*, dan *javascript* seperti bangunan rumah, perabotan di dalamnya, serta lampu dan pintu-pintu. Namun sekarang, *javascript* juga mulai beralih kepada *crossplatform* yaitu aplikasi multi platform seperti ke sistem operasi *Android* dan *IOS* selain daripada *WEB* seperti sebelumya. Perkembangan *javascript* juga memungkinkan programmer untuk membuat aplikasi desktop.

Seperti yang telah disinggung sebelumnya bahwa *javascript* yaitu bahasa pemrograman yang dikembangkan oleh Netscape. Proses desainnya memerlukan waktu selama sepuluh hari. Nama awal dari *javascript* yaitu Mocha, lalu Mona, Live script, baru kemudian *javascript*. Awalnya bahasa pemrograman *Javascript* yang dibuat bulan September 1995 ini hanya dipakai oleh kalangan Netscape, fungsinya pun terbatas tidak seperti saat ini. Namun seiring berjalannya waktu, *javascript* terus dikembangkan hingga bisa seperti sekarang ini. Mulanya di tahun 1996 *javascript* disebut sebagai *ECMAScript*. Hal tersebut berlanjut hingga tahun 1998. Di tahun 1998 dimunculkan *ECMAScript* 2 dan tahun 1999 muncul *ECMAScript* 3. *ECMAScript* terus dikembangkan hingga akhirnya menjadi *JavaScript* yang ada saat ini. *Javascript* hampir digunakan oleh 92 persen *website* yang ada pada tahun 2016. Untuk saat ini, mungkin jumlahnya lebih dari itu.

Namun terdapat web browser yang menawarkan pilihan bagi para pengguna untuk menentukan apakah ingin menonaktifkan atau tidak. Jika tidak diaktifkan maka beberapa fitur tidak akan bisa diproses ataupun tampil. Anda para awam pun pasti bisa memahami hal ini. Jika anda sering berselancar di internet, pasti ada *website* yang menggunakan javascript yang meminta anda untuk mengaktifkannya. Hal tersebut lah yang dimaksud dalam poin ini. Jadi memang ada web browser yang menawarkan pilihan aktif atau tidaknya *JavaScript*.

JavaScript ialah salah satu bahasa pemrograman yang memiliki banyak kelebihan. Kelebihan yang dimaksud antara lain, lebih mudah dipelajari jika dibandingkan dengan bahasa pemrograman yang lainnya. Penanganan dan pencarian kesalahan ataupun error juga lebih mudah. Anda juga tidak membutuhkan compiler sebab web browser bisa menginterpretasikannya dengan *HTML*. Mengenai memori yang digunakan, *javascript* tergolong kecil, ringan, dan lebih cepat dibanding bahasa pemrograman yang lainnya. Kelebihan-kelebihan tersebut tentu saja menjadikan *javascript* sebagai salah satu bahasa pemrograman favorit. Setiap hal tentu memiliki kekurangan, begitu juga dengan *javascript*.

E. PHP

Menurut (Arief, 2011) *PHP* adalah sebuah peranti yang biasa dipakai untuk membuat aplikasi *WEB*. Aplikasi *WEB* adalah aplikasi yang dijalankan melalui *browser*.

PHP adalah bahasa *server*-side –*script*ing yang menyatu dengan *HTML* untuk membuat halaman *WEB* yang dinamis. Karena *PHP* merupakan *server*-side-*script*ing maka sintaks dan perintah-perintah *PHP* akan dieksekusi di *server* kemudian hasilnya akan dikirimkan ke *browser* dengan format *HTML*. Dengan

demikian kode program yang ditulis dalam *PHP* tidak akan terlihat oleh *user* sehingga keamanan halaman *WEB* lebih terjamin. *PHP* dirancang untuk membuat halaman *WEB* yang dinamis.

Kelebihan-kelebihan *PHP* adalah sebagai berikut:

a. *Script*(kode program) terintegrasi dengan file *HTML*, sehingga *developer* (pengembang) bisa berkonsentrasi *Langsung* pada penampilan *WEB* nya.
b. Tidak ada proses *compiling* dan *linking*.
c. Berorientasi obyek (*object oriented*).
d. *Sintaksis* program mudah dipelajari.
e. Integrasi yang sangat luas ke berbagai *server Database*.

F. HTML

HTML (*Hyper Text Markup Language*) adalah bahasa yang digunakan untuk menulis halaman *WEB* dan menampilkan berbagai informasi di dalam sebuah *browser* internet (Jubilee, 2011). *HTML* ini merupakan pengembangan dari standar pemformatan dokumen teks yaitu *Standard Generalized Markup Language* (*SGML*). *HTML* saat ini merupakan standar Internet yang didefinisikan dan dikendalikan penggunaannya oleh *World Wide WEB Consortium* (*W3C*).

HTML dibuat oleh Tim Berners-Lee ketika masih bekerja untuk CERN (*Conseil Europeene Pour La Recherche Nuclaire*) dan dipopulerkan pertama kali oleh *Browser Mosaic* pada tahun 1989 sebagai cara sederhana namun efektif untuk mengkodekan dokumen elektronik. Sebetulnya, tujuan awal dari *WEB browser* adalah untuk melayani sebagai pembaca untuk dokumen berformat *HTML*. Namun, dua dekade kemudian *browser* sendiri telah menjadi sebuah portal ke dunia media online. Itulah sebabnya versi 2 *HTML* yang selanjutnya

tidak hanya sebagai revisi *HTML*, tetapi juga sebagai standar yang *komprehensif* untuk bagaimana halaman *WEB* bekerja.

G. Database

Database (Nugroho, 2008) adalah suatu pengelolaan data dalam pengingat eksternal (misalnya *harddisk*) yang memungkinkan seseorang dengan mudah menyimpan data dan sekaligus menggunakannya ketika memerlukannya. *Software* yang digunakan untuk mengelola *database* antara lain yaitu *MySQL*, *Access*, dan *Oracle*.

Basis data (*Database*) adalah kumpu*lan* data yang disimpan secara sistematis di dalam komputer yang dapat diolah atau dimanipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi. Pendefinisian basis data meliputi spesifikasi berupa tipe data, struktur data dan juga batasan-batasan pada data yang kemudian disimpan.

Basis data merupakan aspek yang sangat penting dalam aplikasi karena berfungsi sebagai gudang penyimpanan data yang akan diolah lebih *lan*jut. Basis data menjadi penting karena dapat mengorganisasi data, menghindari duplikasi data, menghindari hubungan antar data yang tidak jelas dan juga update yang rumit.

Proses memasukkan dan mengambil data ke dan dari media penyimpanan data memerlukan perangkat lunak yang disebut dengan sistem manajemen basis data (*Database management system*). *DBMS* merupakan sistem perangkat lunak yang memungkinkan pengguna basis data (*Database user*) untuk memelihara, mengontrol dan mengakses data secara praktis dan efisien.

Menururut jenisnya Basis data (*Database*) dapat dibagi menjadi :

1. *Basis data flat-file*.

   Basis data ini ideal untuk data berukuran kecil dan dapat dirubah dengan mudah. Pada dasarnya, *Basis data flat-file*tersusun dari sekumpu*LAN* string dalam satu atau lebih file yang dapat diurai untuk mendapatkan informasi yang disimpan. *Basis data flat-file* cocok untuk menyimpan daftar atau data yang sederhana dan dalam jumlah kecil. *Basis data flat-file* akan menjadi sangat rumit apabila digunakan untuk menyimpan data dengan struktur kompleks walaupun dimungkinkan pula untuk itu.

   Beberapa kendala dalam menggunakan basis data jenis ini adalah rentan pada korupsi data karena tidak adanya penguncian yang melekat ketika data digunakan atau dimodifikasi dan juga adanya duplikasi data yang mungkin sulit dihindari. Salah satu tipe *Basis data flat-file* adalah file CSV yang menggunakan pemisah koma untuk setiap nilainya.

2. *Basis Data Relasional*

   Basis data ini mempunyai struktur yang lebih logis terkait cara penyimpanan. Kata "*relasional*" berasal dari kenyataan bahwa tabel-tabel yang ada di basis data *relasional* dihubungkan satu dengan lainnya. Basis data *relasional* menggunakan sekumpu*LAN* tabel dua dimensi yang masing-masing tabel tersusun atas baris (*tupel*) dan kolom (atribut).

Untuk membuat hubungan antara dua atau lebih tabel, digunakan key (atribut kunci) yaitu *primary key* di salah satu tabel dan foreign key di tabel yang lain. Saat ini, basis data *relasional* menjadi pilihan utama karena keungguLANnya. Program aplikasi untuk mengakses basis data *relasional* menjadi lebih mudah dibuat dan dikembangkan dibandingkan dengan penggunaan *Basis data flat-file*.

Beberapa kekurangan yang mungkin dirasakan di basis data jenis ini adalah implementasi yang lebih sulit untuk data dalam jumlah besar dengan tingkat kompleksitasnya yang tinggi. Selain itu, proses pencarian informasi juga menjadi lebih lambat karena perlu menghubungkan tabel-tabel terlebih dahulu apabila datanya tersebar di beberapa tabel.Namun, terlepas dari beberapa kekurangannya, basis data *relasional* telah digunakan secara luas saat ini.

H. PHPMYADMIN

PHPMYADMIN adalah perangkat lunak bebas yang ditulis dalam bahasa pemrograman *PHP* yang digunakan untuk menangani administrasi *MySQL* melalui *website* Jejaring Jagat Jembar (*World Wide Web*) . *phpMyAdmin* mendukung berbagai operasi *MySQL*, diantaranya (mengelola basis data, tabel-tabel, bidang (*fields*), relasi (*relations*), indeks, pengguna (*users*), perizinan (*permissions*), dan lain-lain).

Pada dasarnya, mengelola basis data dengan *MySQL* harus dilakukan dengan cara mengetikkan baris-baris perintah yang sesuai (*command line*) untuk setiap maksud tertentu. Jika seseorang ingin membuat basis data (*database*), ketikkan baris perintah yang sesuai untuk membuat basis data. Jika seseorang menghapus tabel, ketikkan baris perintah yang sesuai untuk menghapus tabel. Hal tersebut tentu saja sangat menyulitkan karena seseorang harus hafal dan mengetikkan perintahnya satu per satu.

Saat ini banyak sekali perangkat lunak yang dapat dimanfaatkan untuk mengelola basis data dalam *MySQL*, salah satunya adalah *phpMyAdmin*. Dengan *phpMyAdmin*, seseorang dapat membuat *database*, membuat tabel,

mengisi data, dan lain-lain dengan mudah, tanpa harus menghafal baris perintahnya.

phpMyAdmin merupakan bagian untuk mengelola basis data *MySQL* yang ada di komputer. Untuk membukanya, buka *browser* lalu ketikkan alamat *http://localhost/phpmyadmin*, maka akan muncul halaman *phpMyAdmin*. Di situ nantinya seseorang bisa membuat (*create*) basis data baru, dan mengelolanya.

Pengembangan *phpMyAdmin* dimulai pada tahun 1998 oleh *Tobias Ratschiller* seorang konsultan *IT*. *Ratschiller* mengerjakan sebuah program bernama *MySQL-Webadmin* dengan bebas, yang merupakan produk dari *Petrus Kuppelwieser*, yang telah berhenti mengembangkannnya pada saat itu. *Ratschiller* menulis kode baru untuk *phpmyadmin*, dan ditingkatkan pada konsep dari proyek *Kuppelwiesser*. *Ratschiller* meninggalkan proyek *phpMyAdmin* pada tahun 2001. Sekarang, Sebuah tim dari delapan pengembang yang dipimpin oleh *Oliver Muller* meneruskan pengembangan *phpMyAdmin* di *SourceForge.net.*

Beberapa fitur dalam phpMyAdmin:

1. Antarmuka berbasis web
2. Dukungan banyak fitur MySQL :

a. Menelusuri dan drop basisdata (database), tabel, pandangan (view), bidang (fields) dan indeks.
b. Membuat, menyalin, drop, dan mengubah nama basis data, tabel, kolom dan indeks.

c. Pemeliharaan server, basis data dan tabel, dengan server konfigurasi.
d. Melaksanakan, mengedit dan penunjuk pernyataan-SQL, bahkan batch-queries
e. Mengelola pengguna MySQL dan hak istimewa.
f. Mengelola prosedur penyimpanan.

3. Impor data dari CSV dan SQL.
4. Ekspor data ke berbagai format: CSV, SQL, XML, PDF, ISO / IEC 26300 - OpenDocument Text dan Spreadsheet, Word, Excel, LATEKS dan lain-lain.
5. Membuat grafik PDF dari tampilan basis data anda.
6. Membuat kompleks query menggunakan Query-by-example. (QBE).
7. Pencarian global dalam basis data.

I. MySQL

MySQL menurut (Arief, 2011) adalah salah satu jenis *Databaseserver* yang menggunakan *SQL* sebagai bahasa dasar untuk mengakses *Database*nya. Dengan menggunakan *scriptPHP* dan *PERLSoftwareDatabase* ini dapat berfungsi atau berjalan pada semua platform sistem operasi yang biasa digunakan (*Windows, Linux, OS/2, berbagai varian Unix*).

Secara garis besar *MySQL* digunakan untuk membuat dan mengelola suatu *Database* secara terstruktur dan otomatis menggunakan suatu Bahasa khusus. Namun lebih jelasnya lagi *MySQL* ini memberikan kemudahan bagi para pengguna yang ingin mengelola suatu data yang berisi informasi secara String

(*text based*) dan dapat diakses secara pribadi maupun untuk umum dalam suatu *WEB*. Hampir semua host atau penyedia *serverWEB* memberikan fasilitas *MySQL* untuk para developer *WEB* yang menginginkan pengelolaan *Database* di *website*nya.

Dalam dunia *website*nya bentuk *interface* atau tatap muka untuk *MySQL* ini sering disebut dengan *PHPMyadmin*. Inilah yang menjadi salah satu faktor adanya keterkaitan antara Bahasa *PHP* dengan *MySQL* terutama dalam hal pengelolaan *WEB*.

Secara umum SQL hanya memiliki 3 komponen penting, yang digunakan untuk mendefinisikan, memanipulasi serta sebagai pengontrol data yang ada pada sebuah database.

Komponen – komponen tersebut diantaranya :

1. *Data Definisi Language (DDL)*

   DDL merupakan bahasa atau perintah pada SQL yang digunakan untuk mendefinisikan data seperti menulis, menghapus dan mengatur atau mengubah data.

   Perintah – perintah yang ada pada komponen DDL yaitu CREATE, DROP, dan ALTER.

   a. CREATE

      Digunakan untuk membuat database, membuat tabel, membuat view dan membuat index.

      Contoh sintak

      Create table namatabel

      (namakolom1 typedata1,

      namakolom2 typedata2,

namakolom3 typedata3)Type data dapat berupa char(), int, numerik dan sebagainya sesuai denga nama kolom.

b. DROP

Drop pada SQL adalah perintah yang di gunakan untuk menghapus data – data yang telah di simpan pada database

Contoh sintak

Drop table

c. ALTER

kegunaan dari perintah alter adalah untuk merubah atribut pada suatu tabel

Contoh sintak

Alter table namatabel

Modify (namakolom typekolom) untuk mengganti type kolom dan nama kolom.alter table namatabel

add (namakolom type kolom[[before, namakolom]]) menambah kolom.alter table namatabel

drop (namakolom typekolom) untuk menghapus kolom pada sebuah tabel.

2. *Data Manipulation Language (DML)*

a. INSERT

insert berfungsi untuk menambah niali pada suatu tabel.

Contoh sintak

insert into namatabel values (‘value1’, ‘value2’,’ …’)keterangan : value adalah nilai yang ingin dimasukkan ke dalam kolom

b. UPDATE

Berfungsi untuk merubah record atau nilai didalam suatu kolom

Contoh sintak

Update namatabel SET namakolom = recordbaru where kondisi keterangan record baru di ubah sesuai dengan nama nilai baru yang akan di ubah

c. *DELETE*

Menghapus record atau nilai didalam suatu kolom pada tabel tertentu

Contoh sintak

DELETE nama kolom from namatabel where kondisi

d. *SELECT*

select adalah perintah yang di gunakan untuk menampilkan nilai atau recor yang di pilih

Contoh sintak

select nama kolom from namatabel

3. *Data Control Language (DCL)*

a. *GRANT*

grant berfungsi untuk memberikan izin akses kepada pengguna

Contoh sintak

Grant privilages on namatabel to nama pengguna artinya memberikan izin akses kepada nama pengguna

b. *REVOKE*

yaitu untuk mencabut izin akses kepada pengguna yang telah di beri izin akses

Contoh sintak

REVOKE privileges ON namatabel from nama pengguna

4. Macam – macam fungsi pada SQL
   a. Tanda *

      digunakan untuk memilih semua isi tabel

      Contoh sintak

      select * from nama tabel

   b. Fungsi WHERE

      berfungsi untuk menentukan suatu pilihan atau kondisi tertentu

      Contoh sintak

      select * from nama tabel where nama kolom = “kondisi”

   c. Fungsi DISTINC

      digunakan untuk menghilangakan nilai ganda pada suatu kolom atau tabel

      Contoh sintak

      select distinct nama kolom from nama tabel

   d. Operator AND

      digunakan untuk memilih nilai yang memiliki kondisi sama antara kondisi1 dan kondisi2

      contoh sintak

      select * from nama tabel where kondisi1=’..’ AND kondisi2 = ‘…’

   e. Operator OR

      digunakan untuk memilih suatu tabel atau kolom dan menampilkan nilai jika salah satu kondisi dari kondisi yang di tentukan memenuhi

      Contoh sintak

select * from nama tabel where kondisi 1='..' OR kondisi 2 = '…'

f. Fungsi NOT

digunakan untuk memilih suatu nilai dari tabel atau kolom dan mengurutkan data tersebut

Contoh sintak

select nama kolom from nama tabel

where namakolom = 'kondisi'

order by nama kolom pengurut desc/asc keterangan : nama kolom pengurut adalah dasar pengurutan

g. Fungsi GROUP BY

berfungsi untuk menelompokkan suatu data, namun pada pada fungsi ini juga biasanya paling banyak melibatkan operator pembanding dan fungsi perhitungan

Contoh sintak

select nama kolom from nama tabel

group by nama kolom yang akan dikelompokkan

having count nama kolom > 4 keterangan having count adalah fungsi yang digunakan untuk mendukung fungsi group by.

J. Metode Pengujian

1. Pengujian Black Box

Pengujian black box adalah metode pengujian perangkat lunak di mana pengujian dilakukan tanpa pengetahuan tentang bagaimana perangkat lunak diimplementasikan secara internal. Artinya, pengujian ini fokus pada fungsi-fungsi eksternal perangkat lunak dan tidak memperhatikan detail

kode atau struktur internalnya. Metode pengujian black box bertujuan untuk mengidentifikasi kesalahan atau cacat dalam perilaku perangkat lunak tanpa harus memahami bagaimana perangkat lunak tersebut bekerja di dalamnya.

Beberapa metode pengujian black box yang umum digunakan meliputi:

a. Pengujian Fungsional: Metode ini memeriksa apakah perangkat lunak berperilaku sesuai dengan spesifikasi fungsionalnya. Ini termasuk pengujian input-output, pengujian kasus pengguna, dan pengujian fungsionalitas spesifik.
b. Pengujian Keamanan: Metode ini fokus pada identifikasi potensi kerentanan keamanan dan ancaman terhadap perangkat lunak, seperti pengujian penetrasi dan pengujian keamanan web.
c. Pengujian Uji Beban: Pengujian ini mengukur kinerja perangkat lunak dengan memberikan beban kerja yang berat dan melihat bagaimana perangkat lunak tersebut menangani situasi tersebut.
d. Pengujian Kesalahan Pengguna: Pengujian ini menguji respons perangkat lunak terhadap kesalahan yang mungkin dibuat oleh pengguna, seperti input yang tidak valid.
e. Pengujian Kompatibilitas: Ini melibatkan pengujian perangkat lunak pada berbagai lingkungan perangkat keras dan perangkat lunak, seperti berbagai sistem operasi atau perangkat seluler.

2. Pegujian White Box

Pengujian dilakukan untuk menjamin operasi-operasi internal sesuai dengan spesifikasi yang telah ditetapkan dengan menggunakan struktur kendali dari prosedur yang dirancang. Metode white Box Digunakan untuk mengetahui cara kerja internal suatu perangkat lunak. Metode perancangan

test case yang menggunakan struktur control dari perancangan procedural untuk mendapatkan test case.

Pengujian white box adalah metode pengujian perangkat lunak di mana pengujian dilakukan dengan memeriksa struktur internal perangkat lunak. Ini berarti bahwa dalam pengujian white box, Anda memiliki pengetahuan tentang bagaimana perangkat lunak diimplementasikan, termasuk kode sumbernya. Metode ini bertujuan untuk memeriksa logika internal, aliran kontrol, dan struktur data perangkat lunak, serta memastikan bahwa kode tersebut memenuhi standar kualitas dan keamanan yang diperlukan.

Beberapa teknik yang digunakan dalam pengujian white box meliputi:

a. Pengujian Unit: Ini adalah pengujian tingkat terendah yang memeriksa setiap unit atau komponen kode secara terpisah untuk memastikan bahwa setiap unit berperilaku sesuai dengan spesifikasi.
b. Pengujian Jalur Pilihan (Path Testing): Teknik ini memeriksa semua jalur eksekusi yang mungkin dalam kode, termasuk jalur yang jarang dieksekusi atau tidak dieksekusi sama sekali.
c. Pengujian Cabang (Branch Testing): Pengujian ini memeriksa setiap percabangan dalam kode untuk memastikan bahwa setiap percabangan telah diuji dan menghasilkan hasil yang benar.
d. Pengujian Integrasi: Ini menguji interaksi antara unit-unit yang berbeda dalam perangkat lunak untuk memastikan bahwa mereka berkomunikasi dengan benar.
e. Pengujian Pemutusan (Fault Injection): Pengujian ini melibatkan penyisipan kesalahan secara sengaja ke dalam kode untuk menguji bagaimana perangkat lunak merespons kesalahan tersebut.

f. Pengujian Kode Sumber Statis: Ini melibatkan analisis statis kode sumber untuk mengidentifikasi potensi masalah seperti variabel yang tidak digunakan, kode mati, atau pelanggaran aturan pemrograman.

K. UML

UML (Unified Modelling Language) menurut menurut Adi nugroho (2010:6), "*Unified Modelling Language* adalah bahasa pemodelan untuk sistem atau perangkat lunak yang berpradigma berorientasi objek". Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupa sehingga lebih mudah dipelajari dan dipahami.

Metode *Unified Modelling Language* (UML) menggunakan tiga bangunan dasar untuk mendeskripsikan sistem atau perangkat lunak yang akan dikembangkan, yaitu:

a. Sesuatu (*things*)

Ada empat *things* dalam *Unified Modelling Language* (UML):

1) *Structural things*, bagian yang relatif statis dapat berupa elemen-elemen yang bersifat fisik maupun konseptual.
2) *Behavorial things,* bagian dinamis biasanya merupakan kata kerja dari model UML yang mencerminkan perilaku sepanjang waktu
3) *Grouping things,* bagian pengorganisasian dalam UML. Dalam penggambaran model UML yang rumit diperlukan penggambaran paket yang menyederhanakan model. Paket-paket ini kemudian dapat didekomposisi lebih lanjut. Paket berguna bagi pengelompokan sesuatu, misalnya model-model serta subsitem-subsistem.
4) *An notational things,* merupakan bagian yang meperjelas model UML. Dapat berisi komentar yang menjelaskan fungsi serta ciri-ciri tiap element dalam model UML

b. Relasi (*relationship*)

Ada empat *relationship* (hubungan) dalam *Unified Modelling Language* (UML):

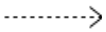
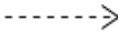
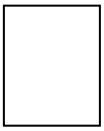
1) Ketergantungan (d*ependency*) adalah hubungan dimana perubahan yang terjadi pada suatu elemn independent akan mempengaruhi elemen yang bergantung padanya.
2) Asosiasi adalah apa dan bagaimana yang mengubungkan antara objek satu dengan yang lainnya. Suatu bentuk asosiasi adalah agregasi yang menampilkan hubungan suatu objek dengan bagian-bagiannya.
3) Generalisasi adalah hubungan dimana objek anak berbagai perilaku dan struktur data dari objek yang ada diatasnya (objek induk). Arah dari objek induk ke objek anak dinamakan spesialisasi sedangkan arah sebaliknya dinamakan generalisasi..
4) Realisasi adalah operasi yang benar-benar dilakukan oleh suatu objek.

c. Diagram, *Unified Modelling Language* (UML) menyediakan Sembilan jenis diagram yang dapat dikelompokkan berdasarkan sifatnya (statis dan dinamis).

1. diagram *use case*

Diagram *use case* menyajikan interaksi antara *use case* dan aktor, dimana aktor dapat berupa orang, peralatan, atau sistem lain yang berinteraksi dengan sistem yang sedang di bangun. *Use case* menggambarkan fungsionalitas sistem atau persyaratan-persyaratan yang harus dipenuhi sistem dari pandangan. Adapun simbol-simbol Use Case Diangram antara lain :

Table 0.1 Simbol *Use Case Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		*Actor*	Menspesifikasikan himpuan peran yang pengguna mainkan ketika berinteraksi dengan *use case*.
2		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (*independent*).
3		*Generalization*	Hubungan dimana objek anak (*descendent*) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
4		*Include*	Menspesifikasikan bahwa *use case* sumber secara *eksplisit*.
5		*Extend*	Menspesifikasikan bahwa *use case* target memperluas perilaku dari *use case* sumber pada suatu titik yang diberikan.
6		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		*System*	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		*Use Case*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor

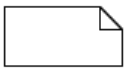
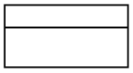
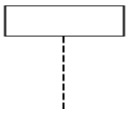
NO	GAMBAR	NAMA	KETERANGAN
9		*Collaboration*	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan prilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		*Note*	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Table 0.2 Simbol *Class Diagram*

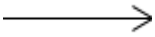
NO	GAMBAR	NAMA	KETERANGAN
1		*Generalization*	Hubungan dimana objek anak *(descendent)* berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (*ancestor*).
2		*Nary Association*	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		*Class*	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		*Collaboration*	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		*Realization*	Operasi yang benar-benar dilakukan oleh suatu objek.

NO	GAMBAR	NAMA	KETERANGAN
6		*Dependency*	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri *(independent)* akan mempegaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya

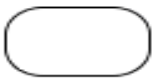
Table 0.3 Simbol *Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		*LifeLine*	Objek *entity*, antarmuka yang saling berinteraksi.
2		*Message*	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		*Message*	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi

Tabel 4. Simbol StateChart Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		*State*	Nilai atribut dan nilai link pada suatu waktu tertentu, yang dimiliki oleh suatu objek.
2		*Initial Pseudo State*	Bagaimana objek dibentuk atau diawali
3		*Final State*	Bagaimana objek dibentuk dan dihancurkan
4		*Transition*	Sebuah kejadian yang memicu sebuah state objek dengan cara memperbaharui satu atau lebih nilai atributnya
5		*Association*	Apa yang menghubungkan antara objek satu dengan objek lainnya.
6		*Node*	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

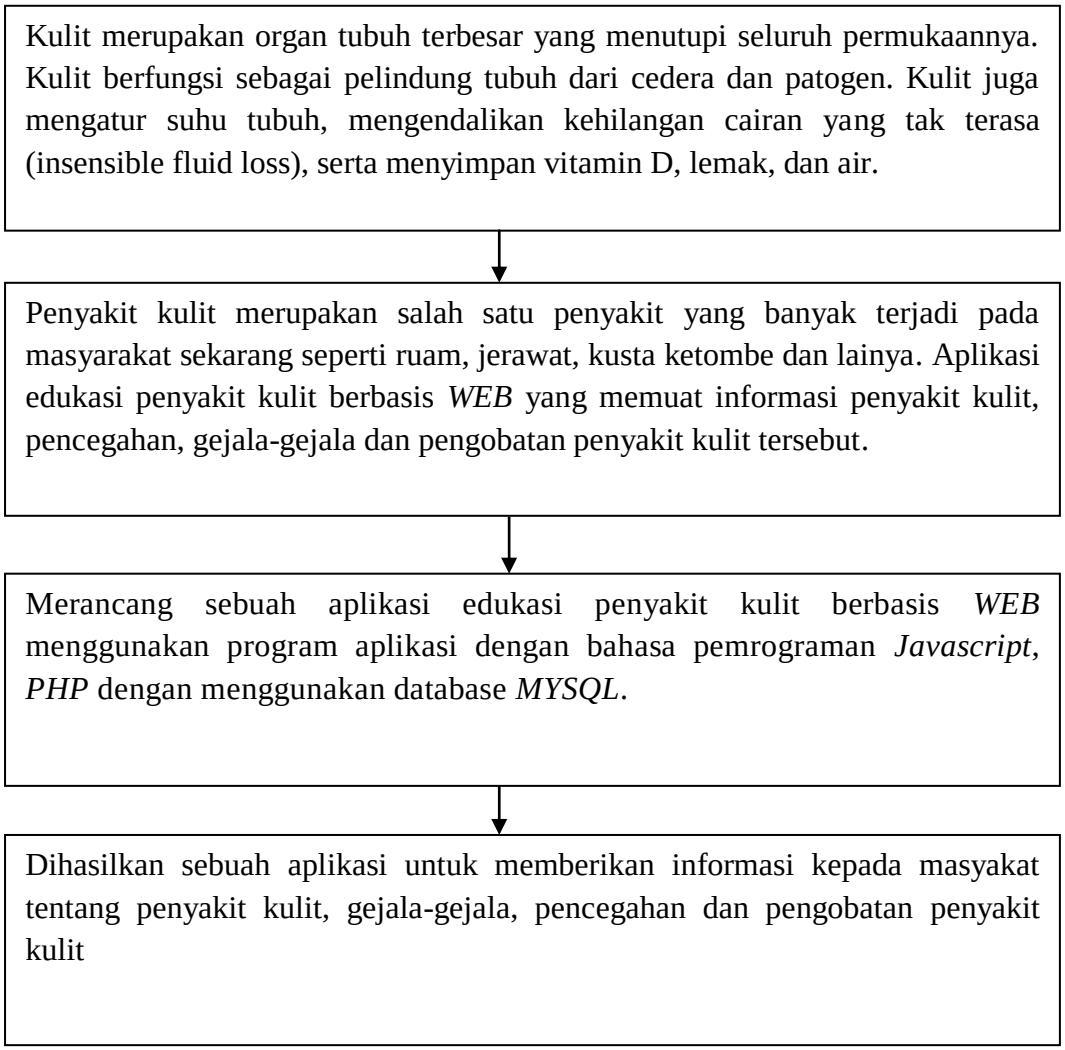
Tabel 5. Simbol Actifity Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		*Actifity*	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		*Action*	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		*Initial Node*	Bagaimana objek dibentuk atau diawali.

NO	GAMBAR	NAMA	KETERANGAN
4		*Actifity Final Node*	Bagaimana objek dibentuk dan dihancurkan
5		*Fork Node*	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

L. Kerangka Pikir

Untuk lebih memperjelas kerangka berpikir yang akan disajikan maka berikut ini digunakan dalam bentuk diagram.



Gambar 0.1 Kerangka Pikir

BAB III

METODE PENELITIAN

A. Jenis Penelitian

Jenis penelitian yang digunakan adalah jenis penelitian Phenomenological research, merupakan salah satu jenis penelitian kualitatif, dimana peneliti melakukan pengumpulan data dengan observasi partisipan untuk mengetahui fenomena esensial partisipan. Jenis penelitian ini merupakan suatu metode atau kegiatan ilmiah yang dilakukan secara sistematis objektif, dan logis dengan mengendalikan atau tanpa mengendalikan berbagai aspek/variable, tentang suatu fenomena, produk, model ataupun fakta yang diteliti.

B. Lokasi dan Waktu Penelitian

Penelitian ini dilakukan di Kota Pare-Pare Sulawesi Selatan, waktu yang dipergunakan untuk pelaksanaan penelitian ini berlangsung selama ± 3 bulan.

Tabel 0.1 Jadwal Pelaksanaan Penelitian

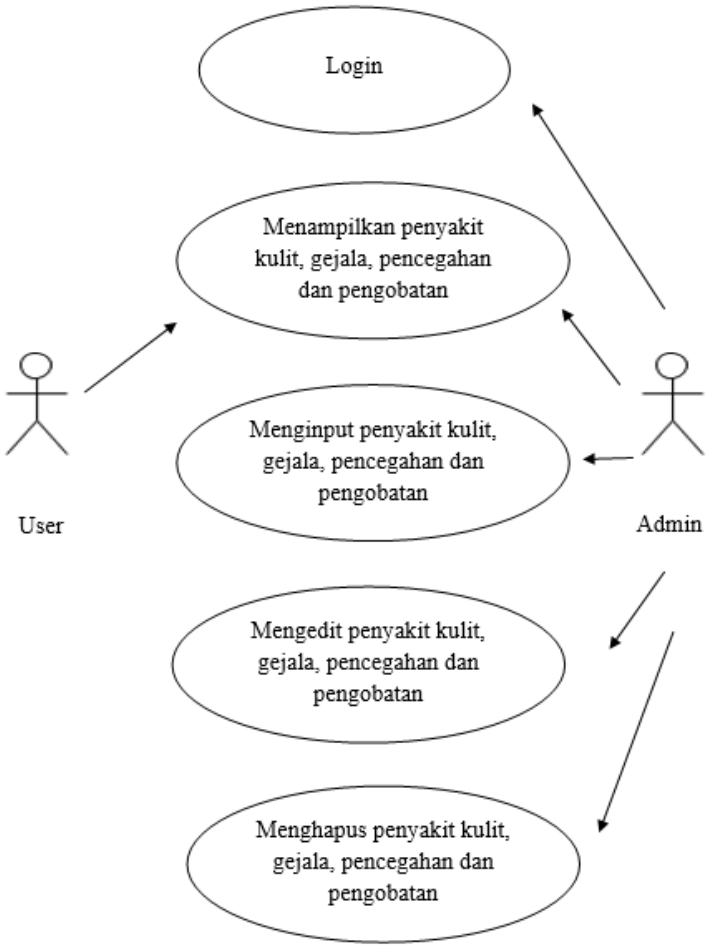
No.	Uraian Kegiatan	Bulan 2022		
		Oktober	November	Desember
1	Studi Literatur			
2	Pembuatan Aplikasi			
3	Pengujian Aplikasi			
4	Hasil Pengujian			

C. Alat dan Bahan Penelitian

Dalam pelaksanaan peneltian ini digunakan beberapa alat dan bahan yang terdiri dari perangkat keras dan perangkat lunak. Perangkat keras yang digunakan terdiri dari *personal computer* (*PC/laptop*) *Laptop-1pv1mjic AMD Athlon gold 3150u, WebCam* dan perangkat lunak yang digunakan adalah Windows 10, *Visual Studio Code*. Sedangkan data yang dibutuhkan adalah buku dan literature berkaitan penyakit kulit, pencegahan, gejala-gejala dan pengobatan.

D. Rancangan Sistem

1. Sistem yang diusulkan



Gambar 0.1 Sistem Yang Dibuat

Diatas merupakan sistem yang diusulkan oleh penulis yanga user dapat menampilkan penyakit kulit, gejala, pencegahan dan pengobatan. Sedangkan admin harus login terlebih dahulu, kemudia dapat menginput, mengedit dan menghapus informasi tentang penyakit kulit yang terdapat pada sistem.

E. Teknik Pengumpulan Data

Pengumpulan data dilakukan untuk mengumpulkan seluruh informasi yang terkait dan mendukung pelaksanaan penelitian penelitian ini.

1. Studi literatur, teknik pengumpulan data yang akan digunakan dalam penelitian ini yaitu studi literatur. Pengumpulan data dengan cara mengambil data, membaca, dan mempelajari literatur dari sumber-sumber seperti buku–buku, skripsi, jurnal yang berhubungan dengan penelitian.
2. Wawancara, metode ini dilakukan kepada narasumber dengan cara mengajukan pertanyaan-pertanyaan yang mendukung permasalahan. Penulis melakukan wawancara dengan mengajukan beberapa pertanyaan yang sederahana namun berkaitan dengan permasalahan yang ada.

Observasi, teknik pengumpulan data menggunakan teknik observasi ini yaitu dengan malakukan pengamatan terhadap penyakit kulit. Hasil dari pengamatan ini akan digunakan sebagai bahan referensi dalam pembuatan Aplikasi Edukasi Penyakit Kulit Berbasis *WEB*.

F. Tahap Penelitian

Tahap penelitian dilakukan dengan cara :

1. Pengumpulan data

Pada tahap ini penulis sedang mengumpulkan beberapa data dan informasi untuk referensi Ketika merancang dan menyelesai kantugas akhir ini.

2. Analisis sistem

Pada tahap ini penulis mengidentifikasi dan menemukan permasalahan yang terjadi serta mencari alternatif solusi yang dapat digunakan untuk menyelesaikan permasalahan tersebut.

3. Desain sistem

Setelah mendefinisikan masalah dan melakukan analisis data, system harus dirancang untuk memfasilitasi proses perakitan.

4. Perakitan

Pada tahap ini, kumpulkan semua komponen yang akan gunakan lalu memulai melakukan perakitan disertai penanaman code.

5. Uji Sistem

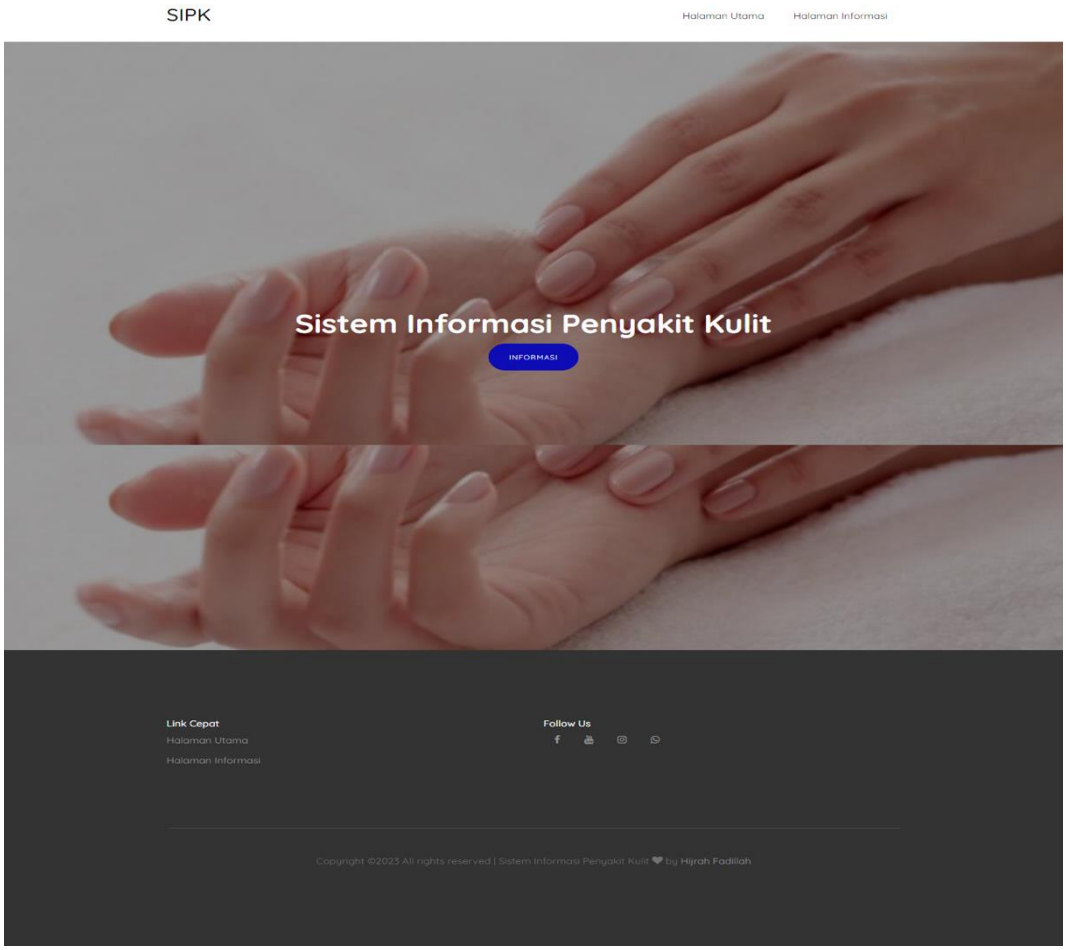
Tes ini dilakukan setelah perakitan benar-benar selesai. Pada pengujian ini seluruh rangkaian/komponen diuji dari rangkaian input hingga rangkaian output untuk mendapatkan hasil.

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

A. Tampilan Aplikasi

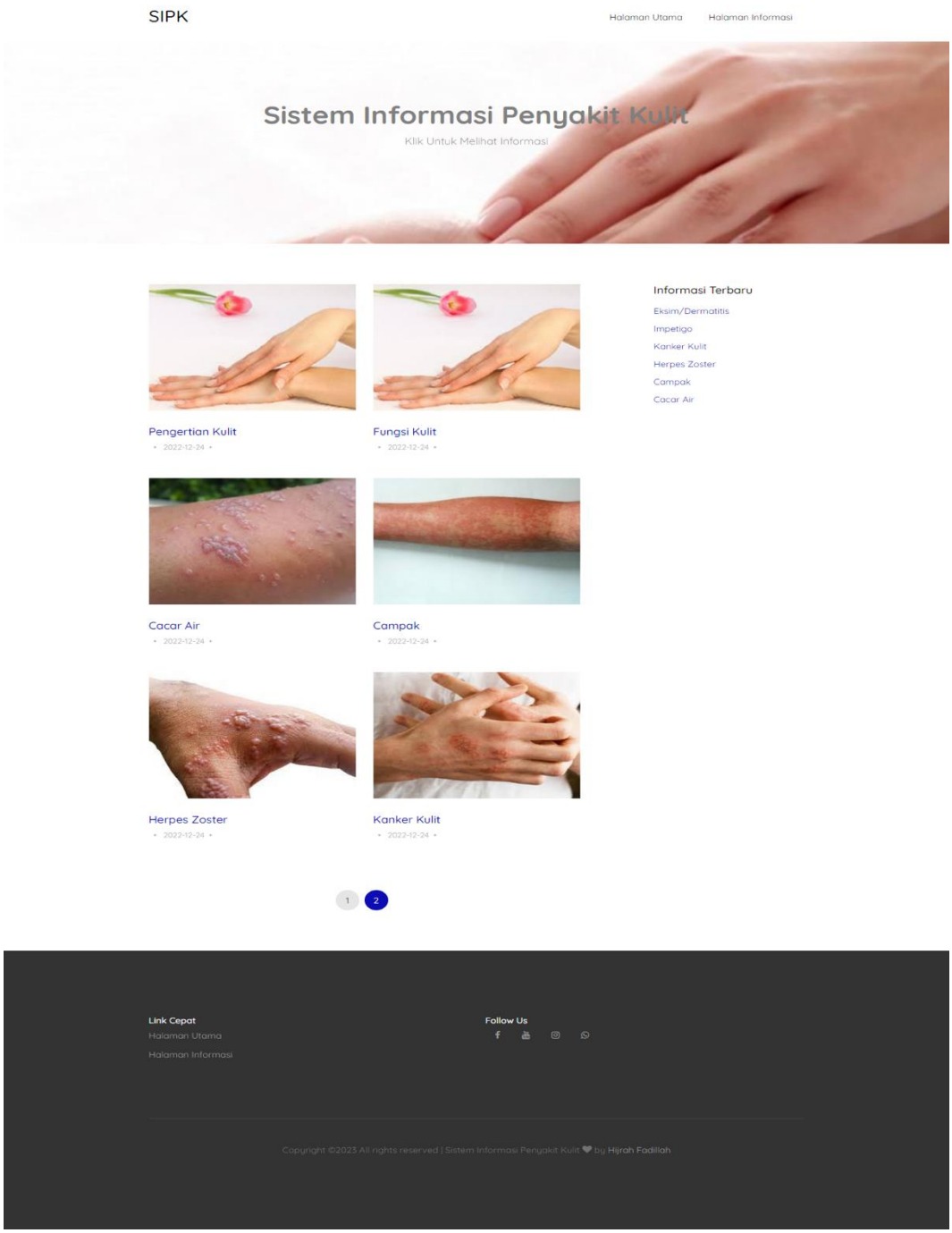
1. Tampilan Awal



Gambar 0.1 Halaman Awal

Diatas merupakan tampilan awal aplikasi yang mana memperlihatkan tombol yang nantinya memuat informasi tentang penyakit kulit yang ada.

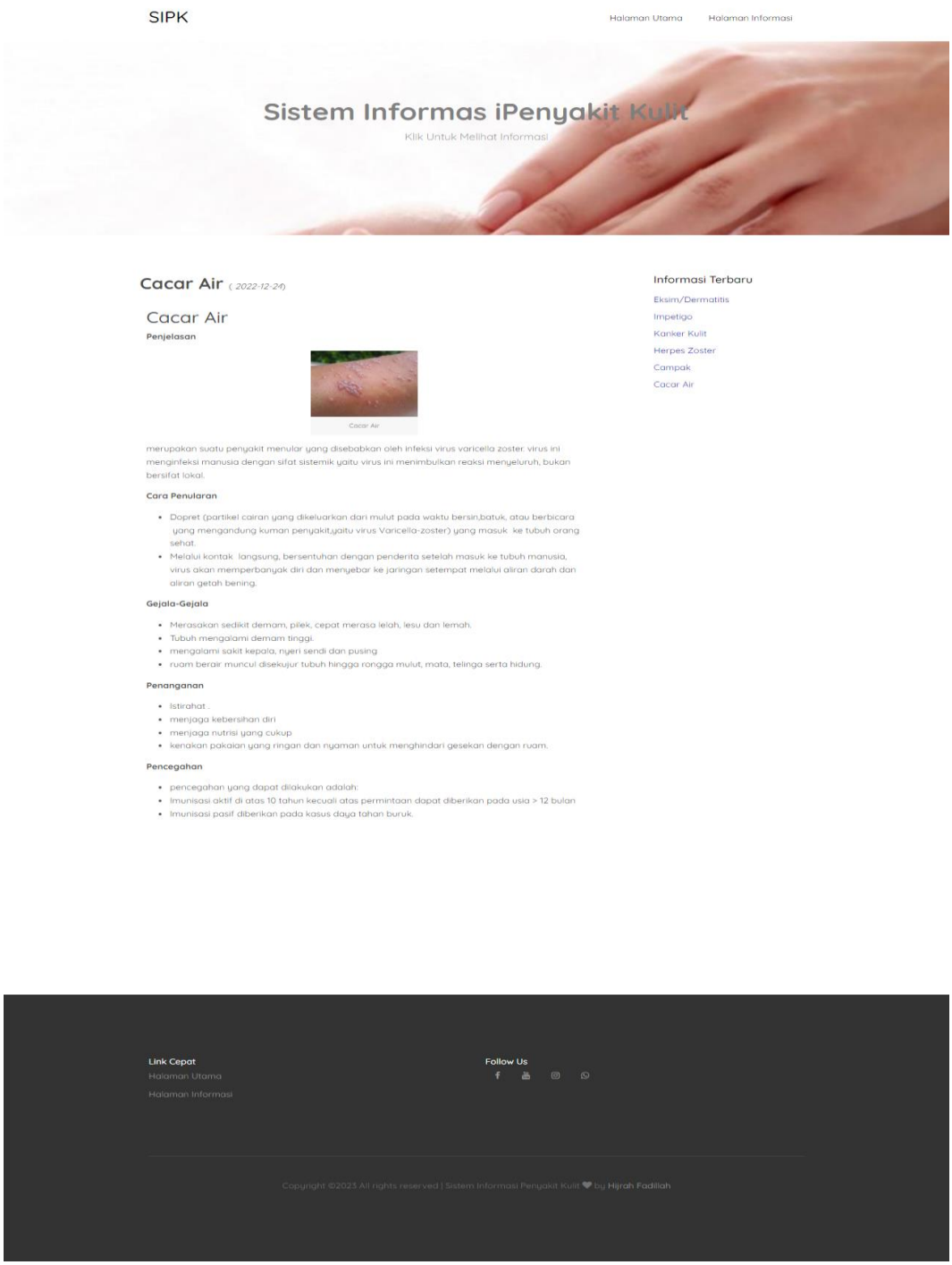
2. Tampilan list informasi



Gambar 0.2 Halaman List Informasi

Berikut diatas merupkan tampilan list informasi penyakit kulit yang ada. Disini user dapat menekan informasi tersbut dan deskripsi informasi akan ditampilakan pada halaman berikutnya.

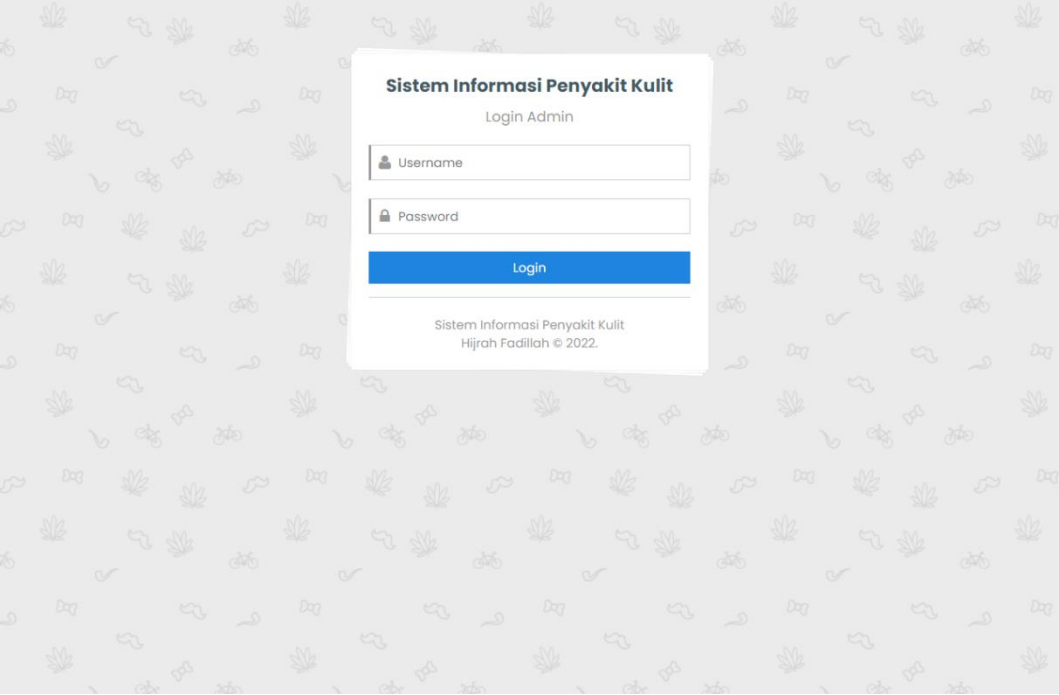
3. Tampilan Deskripsi Penyakit Kulit



Gambar 0.3 Halaman Deskripsi Penyakit Kulit

Diatas merupakan halaman deskripsi penyakit kulit yang memuat informasi penyakit kulit, gejala-gejala , pencegahan dan pengobatan penyakit yang dipilih oleh user.

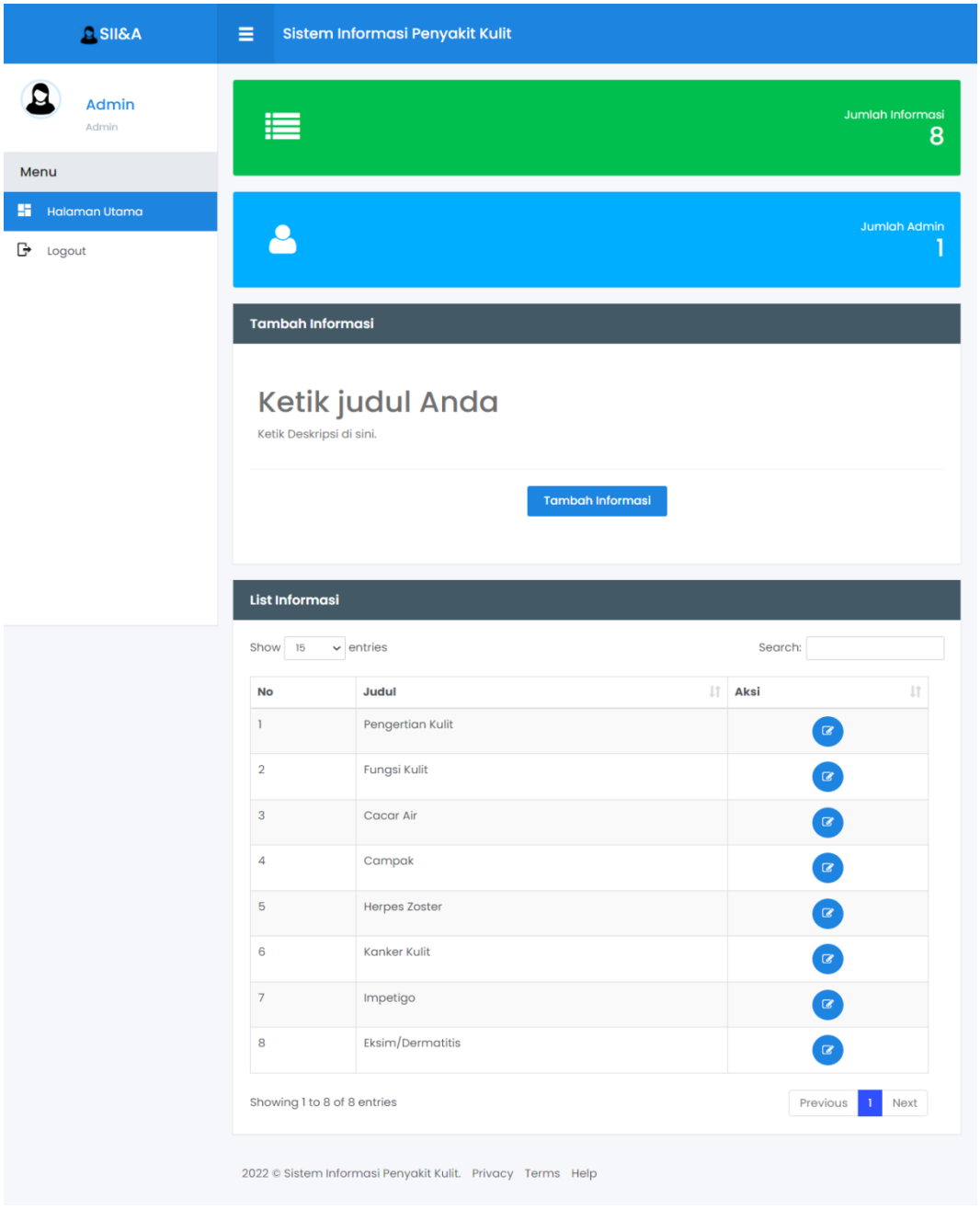
4. Halaman Login



Gambar 0.4 Halaman Login

Diatas merupakan halaman login untuk admin. Yang mana admin harus memasukkan username dan password yang benar untuk bisa mengakses halaman seterusnya.

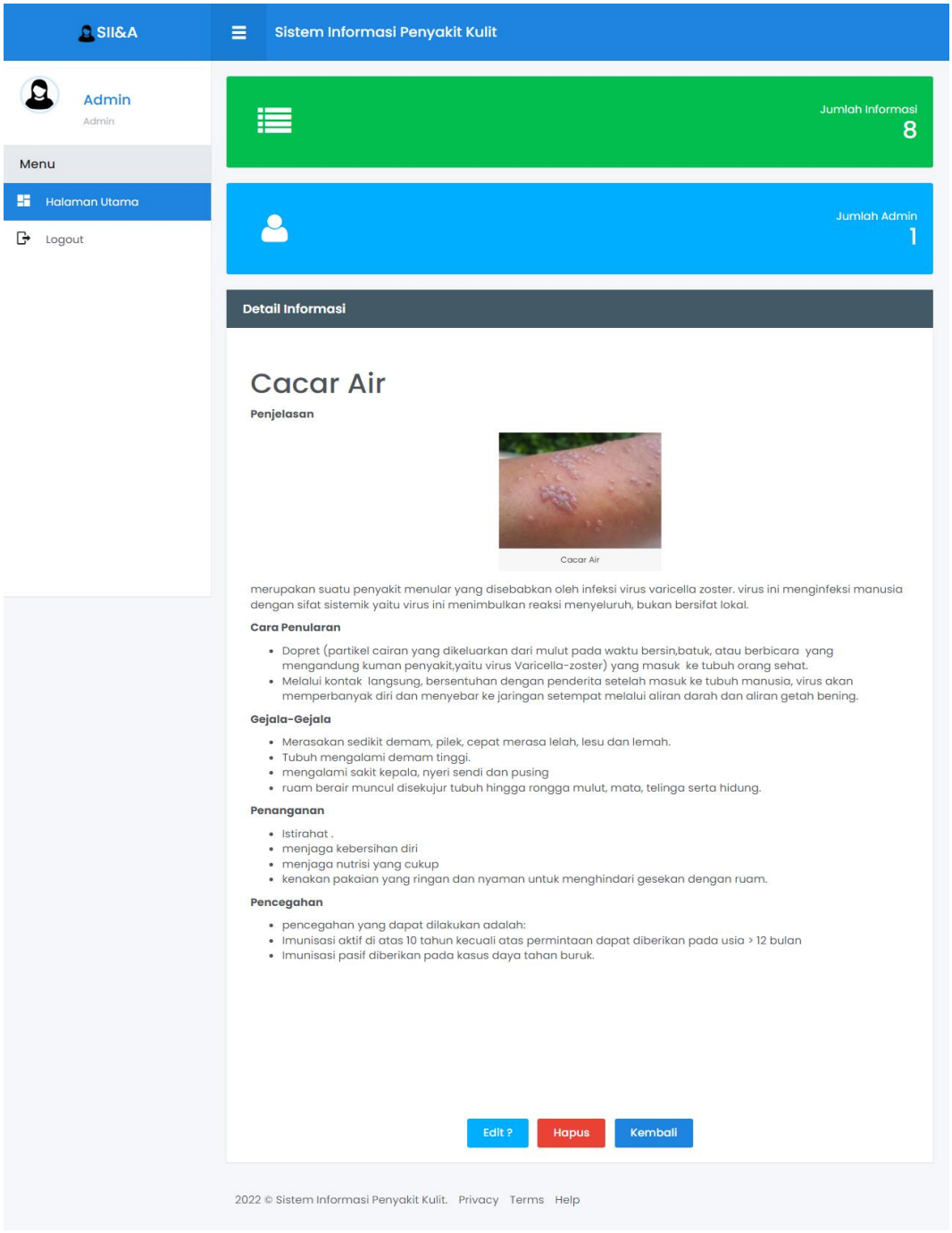
5. Halaman Utama Admin



Gambar 0.5 Halaman Utama Admin

Diatas merupakan halaman utama setelah admin sukses login. Disini admin dapat melihat jumlah informasi, list informasi, dan menginput informasi baru berkaitan penyakit kulit

6. Halaman Detail Admin



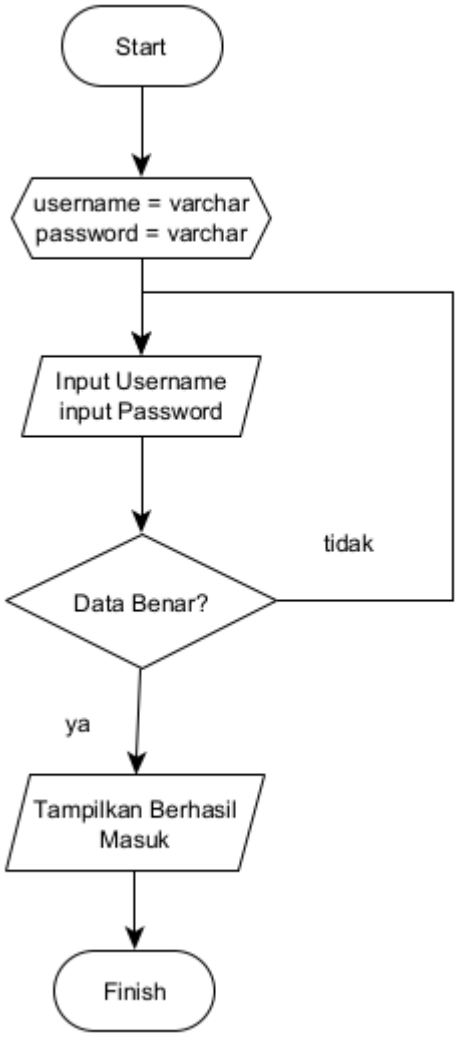
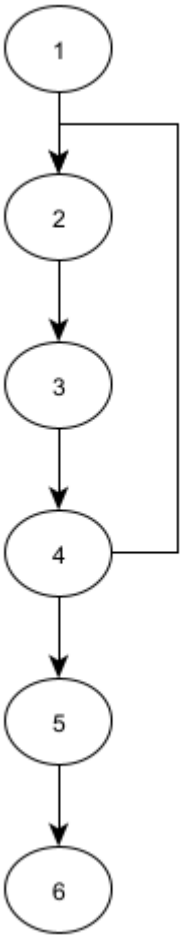
Gambar 0.6 Halaman Detail Admin

Diatas merupakan halaman detail setelah salah satu informasi dipilih oleh admin sebelumnya, disini admin dapat melihat, mengedit dan menghapus informasi tersebut.

B. Pengujian White Box

1. W*hite Box Login*

Tabel 0.1 *White Box Login*

Flowchart	*Flowgraph*
	

Menghitung jumlah *Region, Cyclomatic Complexity* dan *Independent Path.*

Independent Path.

1) *Path* 1 = 1-2-3-4-5-6

2) *Path* 2 = 1-2-3-4-2

Cyclomatic Complexity.

V(G) = E - N + 2

= 6 - 6 + 2 = 2

V(G) = P + 1

= 1 + 1

= 2

Region = 2

Grafik Matriks

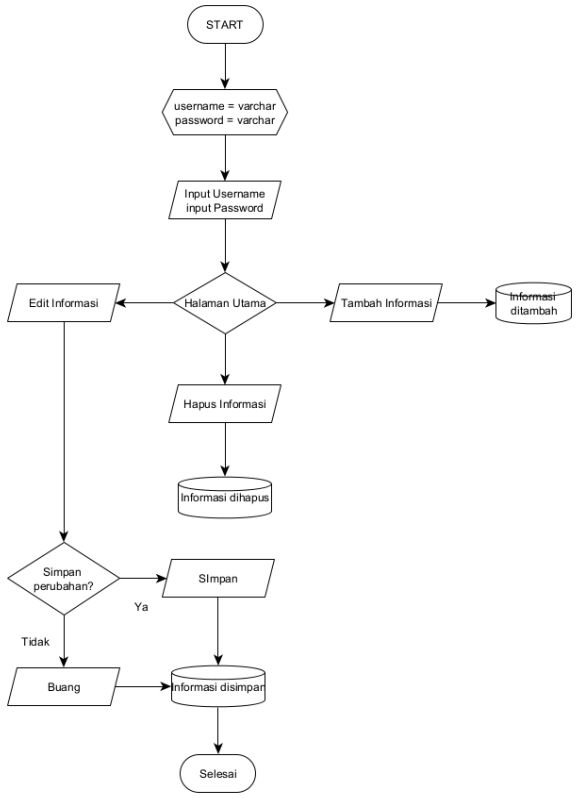
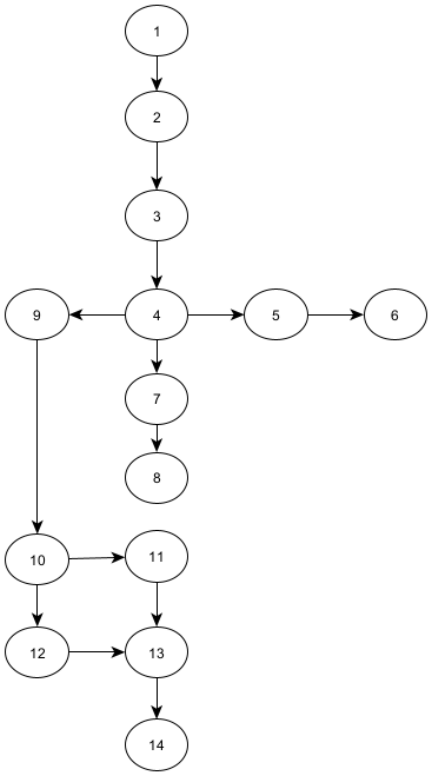
Tabel 0.2 Grafik Matriks L*ogin*

	1	2	3	4	5	6	E-1
1		1					1 - 1 = 0
2			1				1 - 1 = 0
3				1			1 - 1 = 0
4		1			1		2 - 1 = 1
5						1	1 - 1 = 0
6							0 - 1 = -1
SUM (E + 1)							1 + 1 = 2

Berdasarkan perhitungan diatas, Region = 2, Independent Path = 2, Cyclomatic Complexity = 2, maka form Login dinyatakan bebas dari kesalahan.

2. *White Box Admin*

Tabel 0.3 White Box Admin

Flowchart	*Flowgraph*
	

Menghitung jumlah *Region*, *Cyclomatic Complexity* dan *Independent Path.*

Independent Path.

1) *Path* 1 = 1-2-3-4-9-10-11-13-14
2) *Path* 2 = 1-2-3-4-9-10-12-13-14
3) *Path* 1 = 1-2-3-4-5-6
4) *Path* 3 = 1-2-3-4-7-8

Cyclomatic Complexity.

$$V(G) = E - N + 2$$

$$= 12 - 11 + 2 = 3$$

$$V(G) = P + 1$$

$$= 2 + 1$$

$$= 3$$

Region = 3

Tabel 0.4 Grafik Matriks Admin

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	E - 1
1		1													1 - 1 = 0
2			1												1 - 1 = 0
3				1											1 - 1 = 0
4					1		1		1						3 - 1 = 2
5						1									1 - 1 = 0
6															0 - 1 = -1
7								1							1 - 1 = 0
8															0 - 1 = -1
9										1					1 - 1 = 0
10											1	1			2 – 1 = 1
11													1		1 - 1 = 0
12													1		1 – 1 = 0
13														1	1 - 1 = 0
14															0 - 1 = -1
SUM (E + 1)															2 + 1 = 3

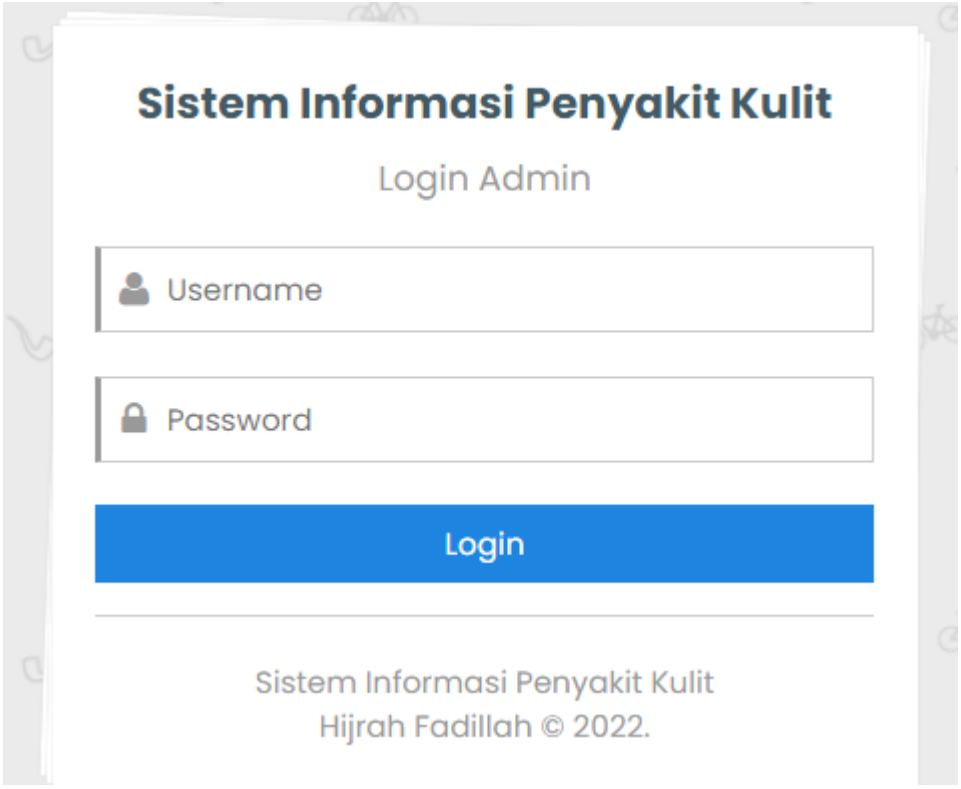
Berdasarkan perhitungan diatas, Region = 3, Independent Path = 3 Cyclomatic Complexity = 3, maka form ini dinyatakan bebas dari kesalahan.

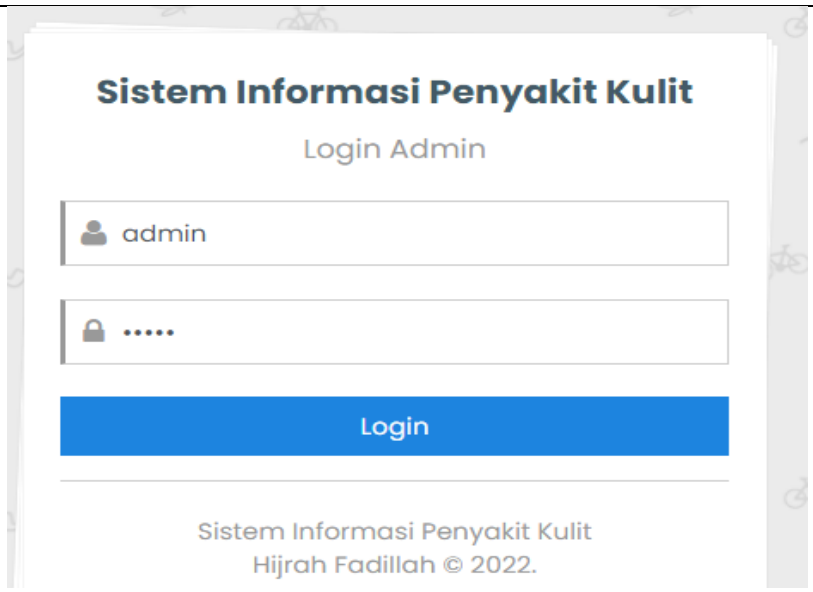
C. Pengujian Black Box

1. *Pengujian Form Login*

Tabel 0.5 Black Box Login

Test Factor	Hasil	Keterangan
Login dengan memasukkan *username* dan *Password*	Berhasil	Jika *Username* dan *Password* benar, maka akan login ke halaman utama




Sistem Informasi Penyakit Kulit
Login Admin
admin
Login
Sistem Informasi Penyakit Kulit
Hijrah Fadillah © 2022.

BAB V

PENUTUP

A. Kesimpulan

Pengetahuan tentang penyakit kulit menjadi penting untuk memahami mekanisme patofisiologis, faktor risiko, dan pengembangan strategi pengobatan yang lebih efektif. Berdasarkan ide tersebut penulis melakukan penelitan sehingga menghasilkan kesimpulan sebagai berikut:

1. Aplikasi edukasi penyakit kulit berbasis web ini berhasil dibangun menggunakan bahasa program HTML, PHP dan Javascript, dengan menggunakan database MySQL sebagai media untuk menyimpan informasi mengenai penyakit-penyakit kulit.
2. Aplikasi ini memiliki hak akses pengguna yang berbeda, admin sebagai administrator dapat menambahkan, mengedit dan menghapus jenis penyakit kulit pada database. Admin menginput username dan password untuk login agar dapat masuk ke halaman administrator.
3. User dapat langusng mengakses informasi mengenai penyakit kulit tanpa harus melakukan login terlebih dahulu. Pada halaman awal aplikasi terdapat *thumbnail* jenis-jenis penyakit kulit, dimana jika *thumbnail* tersebut di klik, maka user akan diberikan informasi detail mengenai penyakit tersebut.

B. Saran

Setelah melakukan penelitian di berkaitan penyakit, maka penulis mengajukan saran untuk pengembangan lebih lanjut sebaiknya aplikasi ini dikembangkan menjadi aplikasi berbasis android agar *user* dan admin dapat mengakses informasi tentang penyakit kulit dimana saja.

DAFTAR PUSTAKA

Arief, M. R. (2011). Pemrograman Web Dinamis menggunakan PHP dan MySQL. Makassar: Penerbit ANDI.

Agung, G. (2019). Lancar Java Dan Javascript. Jakarta: Jublilee Enterprise.

Agung, L. (2010). Aplikasi Pemrograman Javascript untuk Halaman Web. Yogyakarta: Andi Offset.

Amman, P., & Offutt, J. (2017). *Introduction to Software Testing.* New York: Cambridge University Press.

Enterprise, J. (2019). Lancar Java dan Javascript. Jakarta: PT. Elex Media Komputindo.

Jarti, N., & Trisno, R. (2021). Sistem Pakar Diagnosa Penyakit Alergi Pada Anak Berbasis WEB. (F. T. Batam, Ed.) Sistem Pakar Diagnosa Penyakit Alergi Pada Anak Berbasis WEB , 3 (12).

Jubilee. (2011). HTML 5 Cara cepat dan mudah menguasai script HTML 5. Yogyakarta: Jubilee Enterprise.

Maharani, A. (2015). Penyakit Kulit, Perawatan, Pencegahan , Pengobatan. (P. P. Press, Ed.) Yogyakarta: Penerbit Pustaka Baru Press.

Nugroho, A. (2008). Menjadi Administrator Basis Data ORACLE 10 g. Bandung: Infromatika Bandung.

Perangin-angin, R. S., & Segala, J. (2021). Klasifikasi Penyakit Kulit Menggunakan Algoritma Naïve Bayes Berdasarkan Tekstur Warna Berbasis Android. (S. P. Jusantara, Ed.) Klasifikasi Penyakit Kulit Menggunakan Algoritma Naïve Bayes Berdasarkan Tekstur Warna Berbasis Android , 4 (2).

Sianipar, R. (2017). Dasar Pemrograman Javascript , Langkah Demi Langkah. Bandung: Penerbit Andi.

Solichin, A. (2010). MYSQL 5 Dari Pemula Hingga Mahir. Jakarta.

Tohari, H. (2018). Astah - Analisis Serta Perancangan Aplikasi Melalui Pendekatan UML. Yogyakarta: Penerbit Andi.

Pezze, M., & Young, M. (2008). *Software Testing and Analysis: Process, Principles and Techniques* . Rosewood Drive, Danvers: John Wiley & Sons, Inc.