

BAB I

PENDAHULUAN

A. Latar Belakang

Tak dapat dipungkiri jika kemajuan teknologi masa kini berkembang sangat pesat. Hal ini dapat dibuktikan dengan banyaknya inovasi-inovasi yang telah dibuat di dunia ini. Dari hingga yang sederhana, hingga yang menghebohkan dunia. Kemajuan teknologi memang sangat penting untuk kehidupan manusia jaman sekarang. Karena teknologi adalah salah satu penunjang kemajuan manusia. Di banyak belahan masyarakat, teknologi telah membantu memperbaiki ekonomi, pangan, komputer, dan masih banyak lagi.

Seiring dengan berkembangnya teknologi informasi menyebabkan semakin banyak orang mengerti akan pentingnya fungsi komputer dalam membantu pekerjaan mereka. Saat ini perkembangan komputer telah membawa perubahan besar dalam berbagai bidang diantaranya bidang Politik, Ilmu Pengetahuan, Ekonomi, Sosial, Budaya, dan Kesehatan. Perkembangan teknologi informasi, selain perkembangan aplikasi desktop pada komputer juga meliputi perkembangan aplikasi mobile. Seperti yang kita ketahui saat ini, kebutuhan manusia tidak pernah terbatas seperti kebutuhan komunikasi salah satunya. Sehingga, handphone yang kita kenal sebagai alat telpon (komunikasi) genggam semakin berkembang pesat dengan aplikasi-aplikasi terbaru dan bermanfaat untuk kebutuhan manusia di saat ini.

Aplikasi berbasis multimedia interaktif sangatlah banyak ragam serta manfaat luar biasa untuk kebutuhan manusia.

Multimedia adalah media penyampaian dengan menggunakan banyak media seperti teks, gambar, animasi, video, suara. Interaktif adalah interaksi antara multimedia itu sendiri dengan pengguna (user). Sebuah multimedia interaktif juga bisa digunakan sebagai media penyampaian pertumbuhan anak.

Jadi Multimedia Interaktif adalah media yang terdiri dari banyak komponen/media yang saling terintegrasi yang mampu untuk berinteraksi dengan penggunanya. Pertumbuhan adalah bertambahnya ukuran dan jumlah sel serta jaringan interselular, sehingga bertambahnya ukuran fisik dan struktur tubuh sebagian atau keseluruhan, maka dapat diukur dengan satuan panjang dan berat. Anak yang sehat akan menunjukkan tumbuh kembang yang optimal. Pertumbuhan anak bisa dilihat dan diukur dari beberapa parameter pertumbuhan. Sekarang ini keberadaan informasi mengenai pertumbuhan dan faktor yang mempengaruhi pertumbuhan anak balita sangatlah kurang. Sehingga jadwal imunisasi serta pengukuran berat dan tinggi badan balita yang biasanya dilakukan dalam rentang waktu satu bulan sekali dengan cara datang langsung ke posyandu terdekat pun sering terlewat oleh para orang tua, dikarenakan sulitnya menyisihkan waktu untuk melakukan hal tersebut.

Berdasarkan latar belakang diatas, maka dari itu penulis akan mengangkat judul “**Aplikasi Multimedia Interaktif Untuk Mengetahui Tumbuh Kembang Anak Berdasarkan Pengelolaan Posyandu Berbasis Web**”.

B. Rumusan Masalah

Adapun rumusan masalah yang akan dibahas adalah Bagaimana membangun aplikasi untuk membantu orang tua memantau tumbuh kembang anak dan melatih pengembangan kognitif anak.

C. Batasan Masalah

Adapun batasan masalah dalam penelitian ini adalah:

1. Aplikasi berisi penghitungan tumbuh kembang anak dan media pembelajaran interaktif melatih pengembangan kemampuan kognitif 0-5 Tahun
2. Menggunakan bahasa pemrograman Action script 2.0 dalam flash.
3. Menggunakan bahasa pemrograman PHP dalam aplikasi web
4. Aplikasi web di gunakan oleh orang tua.
5. Aplikasi multimedia interaktif digunakan oleh batita dan balita dengan umur 0-5 Tahun dengan di dampingi oleh orang tua

D. Tujuan Penelitian

Tujuan dari pembuatan aplikasi ini adalah untuk merancang dan membuat sebuah aplikasi berbasis web dan media pembelajaran interaktif menggunakan bahasa pemrograman PHP dan *Action Script* 2.0 dan membantu orang tua dalam memantau pertumbuhan anak serta memberikan media belajar interaktif bagi anak usia balita dan batita.

E. Manfaat Penelitian

Adapun manfaat penelitian yaitu sebagai berikut:

1. Aspek Akademis

- a. Sebagai bahan masukan bagi para pengkaji ilmu komunikasi di bidang *Public Relations* yang berminat untuk meneliti masalah yang sama.
 - b. Sebagai langkah awal bagi peneliti untuk dapat mengembangkan teori-teori yang didapat selama ini dan menambah Ilmu khususnya tentang Aplikasi multimedia berbasis *Website*.
2. Aspek Praktis
- a. Sebagai masukan dan bahan informasi bagi orang tua yang berada di kota Parepare.
 - b. Bagi peneliti diharapkan dapat memberikan pengalaman dan menerapkan ilmu pengetahuan yang diterima selama mengikuti perkuliahan.
 - c. Sebagai tugas akhir untuk mendapatkan gelar sarjana strata 1 (S.1) pada Fakultas Teknik Informatika

BAB II

TINJAUAN PUSTAKA

A. Tinjauan Teori

1. Pengertian Perancangan

Perancangan merupakan salah satu hal yang penting dalam membuat program. Adapun tujuan dari perancangan ialah untuk memberi gambaran yang jelas lengkap kepada pemrogram dan ahli teknik yang terlibat. Perancangan harus berguna dan mudah dipahami sehingga mudah digunakan. Perancangan adalah Sebuah Proses untuk mendefinisikan sesuatu yang akan dikerjakan dengan menggunakan teknik yang bervariasi serta di dalamnya melibatkan deskripsi mengenai arsitektur serta detail komponen dan juga keterbatasan yang akan dialami dalam proses pengerjaannya. Menurut Pressman (2009) perancangan atau rancang merupakan serangkaian prosedur untuk menterjemahkan hasil analisa dan sebuah sistem ke dalam bahasa pemrograman untuk mendeskripsikan dengan detail bagaimana komponen-komponen sistem di implementasikan.

2. Aplikasi

Aplikasi berasal dari kata *application* yaitu bentuk benda dari kata kerja *to apply* yang dalam Bahasa Indonesia berarti pengolah. Secara istilah, aplikasi komputer adalah suatu sub kelas perangkat lunak komputer yang menggunakan komputer langsung untuk melakukan suatu tugas yang diinginkan pemakai. Menurut Kamus Besar Bahasa Indonesia (KBBI, 2015)

Aplikasi adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu. Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari pengguna. Sementara menurut Hartono (2021) aplikasi adalah penggunaan dalam suatu komputer, instruksi (*instruction*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses input menjadi output.

3. Multimedia

Multimedia berasal dari kata multi dan media. Multi dapat berarti banyak, sedangkan media dapat berarti sebagai tempat, sarana atau alat yang digunakan untuk menyimpan suatu informasi. Multimedia adalah pemanfaatan perangkat lunak pada komputer untuk membuat dan mengkombinasikan teks, bunyi, *video*, gambar dan animasi dengan menggunakan beberapa *link* (penghubung) dan *tool* (alat) yang memungkinkan bagi pengguna dapat melakukan navigasi, interaksi, berkreasi dan berkomunikasi.

Penyajian multimedia dengan mengkombinasikan beberapa atau seluruh elemen multimedia tersebut, akan menjadikan sebuah informasi dalam bentuk multimedia yang dapat diterima dengan baik oleh indera pendengaran maupun indera penglihatan dan lebih mendekati bentuk asli dalam dunia sebenarnya. (Madiun, 2020).

a. Jenis Multimedia

Dalam pengelompokannya, multimedia terbagi menjadi beberapa macam jenis. Penggunaan jenis multimedia tersebut, antara lain :

1) Multimedia Linier

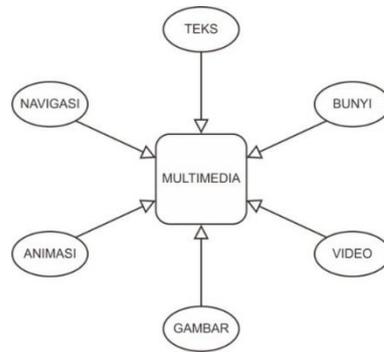
Adalah suatu multimedia yang tidak dilengkapi oleh pengontrol apapun, multimedia ini berjalan secara sekuensial (berurutan), contohnya adalah televisi dan film

2) Multimedia Interaktif

Sudah dilengkapi oleh pengontrol yang dapat dioperasikan oleh pengguna, sehingga pengguna dapat memilih apa yang dikehendaki untuk proses interaksi. Contohnya adalah permainan dan pembelajaran interaktif.

b. Elemen Penunjang Multimedia

Multimedia adalah suatu istilah generik bagi media yang terdiri dari berbagai macam media, baik untuk tujuan pembelajaran maupun bukan. Elemen penunjang pada multimedia meliputi teks, bunyi, *video*, gambar, animasi dan navigasi. Elemen penunjang multimedia dapat dilihat pada gambar 2-1.



Gambar 2. 1 Elemen Penunjang Multimedia

c. Teks

Teks adalah media paling dasar yang bersifat sederhana dan efektif dalam proses penyampaian suatu informasi, media dengan menggunakan teks membutuhkan tempat penyimpanan data yang paling kecil dibandingkan dengan menggunakan media lainnya. Secara umum, teks terbagi menjadi beberapa macam, antara lain :

1) Teks Cetak

Merupakan media yang tercetak di atas kertas.

2) Teks Hasil *Scan*

Merupakan media yang tercetak di atas kertas dan dikonversikan dengan menggunakan komputer dalam bentuk gambar *digital* melalui mesin *scanner*.

3) Teks *Digital*

Media yang dibuat dengan menggunakan perangkat lunak pada komputer untuk membuat dan melakukan pengetikan teks tersebut.

d. *Hyper Text*

Merupakan media penghubung pada sebuah teks yang pada umumnya dikombinasikan dengan tombol untuk dapat terhubung dengan informasi lainnya.

e. *Video*

Bentuk rekaman gambar hidup atau gambar bergerak yang saling berurutan. Dalam fungsinya, terdapat dua macam jenis *video*, yaitu *video analog* dan *video digital*. *Video analog* dibentuk oleh deretan sinyal elektrik (gelombang analog) yang direkam dengan menggunakan kamera dan dipancarluaskan melalui gelombang di udara. Sedangkan *video digital* dibentuk oleh sederetan sinyal *digital* yang berbentuk bilangan biner (0 dan 1).

Pada *video digital* terdapat tiga komponen utama yang terbentuk, yaitu *frame rate*, *frame size* dan *datatype*. Pada *frame rate* menggambarkan berapa kali bingkai gambar yang dapat dimunculkan pada setiap detiknya, sementara *frame size* merupakan ukuran fisik yang sebenarnya dari setiap bingkai gambar yang telah disusun dan *datatype* menentukan seberapa banyak perbedaan warna yang dapat dimunculkan pada saat bersamaan.

Dalam penggunaannya, *video* dapat dimanfaatkan untuk memaparkan keadaan nyata dari suatu proses, fenomena atau kejadian yang dapat terintegrasi dengan media lain pada teks atau gambar.

f. Gambar

Gambar merupakan elemen yang dapat membantu menyampaikan suatu informasi dengan lebih mudah, menarik perhatian dan mengurangi rasa kebosanan. Dalam multimedia, gambar dapat diolah menjadi suatu proses penyampaian informasi dengan cara memahaminya, gambar juga dapat menjadi suatu penghubung kepada informasi yang lainnya. Ketika pengguna melihat sebuah objek gambar, pengguna akan mendapatkan informasi dalam jumlah yang lebih besar jika dibandingkan dengan hanya membacanya, misalnya pada sebuah karya seni lukis, seni grafis dan karya fotografi. Secara umum terdapat beberapa jenis macam gambar, antara lain

1) Gambar Vektor

Merupakan objek gambar yang terbentuk dari sekumpulan kombinasi *node* (titik) dan *line* (garis) yang diperoleh dari suatu rumusan matematika, aljabar linier maupun geometri. Vektor menggambarkan sebuah objek gambar yang disesuaikan dengan perhitungan geometri (posisi sumbu x dan y pada koordinat tertentu).

Sebagai contoh, sebuah objek gambar vektor yang dibentuk dengan proses perhitungan untuk membentuk suatu kurva, menentukan posisi dengan diisi kombinasi warna tertentu. Gambar tersebut secara fleksibel dapat diubah ukurannya, dipindah atau bahkan diubah warnanya tanpa merusak kualitas gambar itu sendiri. Faktor yang menyebabkan kualitas gambar vektor tetap terjaga adalah karena gambar vektor tidak tergantung pada resolusi.

Gambar vektor lebih cocok untuk produksi gambar-gambar yang memiliki tepi tajam. Gambar vektor ini dapat dicetak dengan resolusi yang berbeda-beda tanpa kehilangan detail dari gambarnya, selain itu gambar vektor mempunyai beberapa kelebihan, antara lain :

- a) Gambar vektor banyak digunakan untuk membuat desain yang membutuhkan soliditas dari bentuk, warnadan fleksibilitas ukuran yang tergantung dari kreatifitas desainer untuk mampu merancang atau membuat suatu objek.
- b) Dapat lebih mengeksplorasi kemampuan dalam membuat bentuk, memilih warna, mengatur *layout* (tata letak) dan lain sebagainya.
- c) Memungkinkan membuat karakter, jati diridari desainer yang tidak dapat ditemui pada hasil karya orang lain, dikarenakan sebuah desain vektor merupakan sebuah objek ilustrasi, imajinasi dan teknik dari desainer itu sendiri, sehingga kecil kemungkinannya untuk dapat menemukan ilustrasi, imajinasi dan teknik yang bisa benar-benar sama.

2) Gambar *Bitmap*

Gambar *bitmap* merupakan rekontruksi dari gambar asli yang tersimpan sebagai serangkaian *pixel* (titik-titik) yang memenuhi bidang *pixel* pada layar komputer. Kelebihannya dalam menggunakan gambar *bitmap* antara lain :

1. Proses pengolahan yang minimal.
2. Lebih cepat untuk ditampilkan, dikarenakan gambar *bitmap* dapat dikirimkan secara langsung dari data ke layar monitor.

Kelemahan dari gambar bitmap adalah :

- a) Desainer tidak dapat memperbesar atau memperkecil resolusi gambar *bitmap*, dikarenakan resolusi gambar bitmap tergantung dari gambar aslinya.
- b) Kapasitas data *bitmap* relatif masih besar.
- c) Apabila gambar *bitmap* dikompresi, maka akan berdampak pada penurunan kualitas gambar dikarenakan pada layar monitor menampilkan gambar dengan cara menggunakan jaringan pada *node* (titik), maka gambar *bitmap* dan gambar vektor akan diperlihatkan sebagai *pixel*. Begitu juga ketika pengguna hendak menampilkan gambar vektor, dimana desainer harus melakukan ekspor dari gambar vektor ke dalam *format* data *.jpg, *.jpeg, *.gif, *.png dan lainnya. Sifat gambar vektor yang telah diekspor tersebut, secara otomatis berubah menjadi tipe gambar *bitmap*, meskipun gambar tersebut dibuat dengan menggunakan perangkat lunak gambar vektor. Perbedaan gambar vektor dan gambar *bitmap* dapat dilihat pada gambar 2-2.



Gambar 2. 2 Perbedaan Gambar Vektor dan Bitmap

g. Konsep Multimedia

Dalam membangun sebuah desain multimedia yang terdiri dari beberapa atau seluruh elemen penunjang multimedia yang saling terintegrasi dan berinteraksi antara satu elemen dengan elemen lain, terdapat beberapa konsep yang harus diperhatikan dalam proses membangun desain multimedia, konsep tersebut antara lain :

1) Komunikatif

Dalam membangun sebuah desain multimedia, peran komunikatif mempunyai pengaruh besar dalam tujuan membangun desain multimedia itu sendiri. Komunikatif dapat diartikan sebagai hubungan antara bahasa penyampaian informasi dengan pengguna yang mengakses informasi tersebut. Contohnya adalah perbedaan informasi yang disampaikan antara desain multimedia pembelajaran dengan desain multimedia untuk surat kabar, sehingga peran komunikatif diperlukan agar informasi yang disampaikan dapat mudah dimengerti, dipahami dan diterima dengan baik.

2) Estetis

Pada desain multimedia perlu memperhatikan estetis multimedia, estetis diperlukan agar desain multimedia tersebut dapat mempunyai nilai ketertarikan oleh pengguna serta dapat meningkatkan daya pakai pengguna, beberapa pendukung peran estetis dalam desain multimedia meliputi seni dan keindahan.

3) Ekonomis

Desain multimedia juga harus memperhatikan faktor ekonomis, artinya ukuran data dan jumlah data dari desain multimedia yang disimpan. Apabila data hasil dari desain multimedia tersebut mempunyai kapasitas yang besar atau jumlah data yang banyak, maka hal tersebut akan memperlambat pengguna dalam melakukan interaksi multimedia dan mungkin membingungkan pengguna yang masih awam, sehingga apabila hasil dari desain multimedia tersebut tidak ekonomis, kemungkinan besar akan ditinggalkan oleh penggunanya.

Agar mendapatkan desain multimedia yang komunikatif, estetis dan ekonomis, memerlukan beberapa pedoman untuk membuat tata letak dari teks, bunyi, video, gambar dan animasi yang dikontrol dengan navigasi. Pedoman yang diperlukan dalam desain multimedia antara lain :

1. Kesatuan

Elemen multimedia yang digunakan, dikombinasikan dengan tata letak, fungsi dan halaman yang ditempatkan sedemikain rupa, sehingga elemen tersebut menjadi suatu kesatuan informasi pada satu halaman informasi dengan halaman informasi yang lain.

2. Keseimbangan

Elemen multimedia yang digunakan, dikombinasikan dengan tata letak, fungsi dan halaman yang ditempatkan sedemikain rupa, sehingga elemen tersebut menjadi informasi yang seimbang pada satu halaman informasi dengan halaman yang lainnya.

3. Kontras

Kontras diperlukan untuk menonjolkan bagian yang dianggap lebih penting dari bagian lainnya. Kontras dapat dinyatakan dengan membedakan ukuran, teks, warna, karakter serta desain pada tata letak.

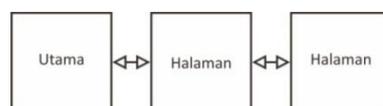
4. Kontinuitas

Informasi yang disampaikan dapat dengan mudah digunakan atau dimengerti oleh pengguna bila mempunyai aliran yang baik, jika mempunyai aliran yang baik, maka aliran informasi dapat dikatakan kontinyu dan harmonis bila pada tampilannya saling berkesinambungan antara bagian satu ke bagian lainnya. Kontinuitas dapat dibuat dengan membangun halaman yang mempunyai karakter, bentuk dan warna, sehingga dapat memberikan rasa berkesinambungan dan harmonis oleh pengguna.

Dalam membangun multimedia ternavigasi yang didukung dengan elemen penunjang multimedia, terbagi menjadi beberapa klasifikasi, antara lain

1. Struktur Linier

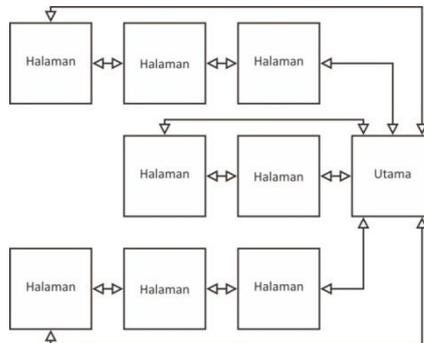
Merupakan struktur sederhana dalam menampilkan aplikasi multimedia dengan dua navigasi maju dan mundur. Struktur linier dapat dilihat pada gambar 2-3.



Gambar 2. 3 Struktur Linier

2. Struktur Menu

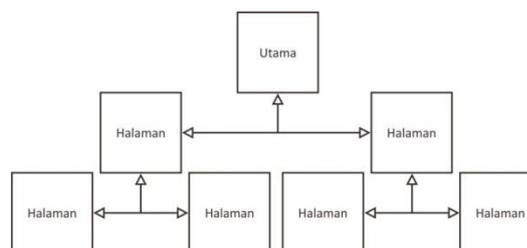
Merupakan sebuah kumpulan navigasi yang dikombinasikan dengan elemen-elemen multimedia. Struktur menu dapat dilihat pada gambar 2-4.



Gambar 2. 4 Struktur Menu

3. Struktur Hirarki

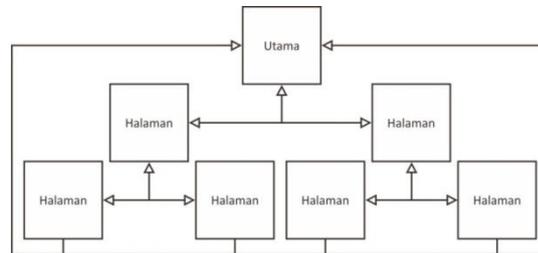
Struktur hirarki berbentuk seperti tangga atau pohon, dimana pada setiap objek menyediakan sebuah navigasi yang memiliki banyak menu. Dalam struktur hirarki, tidak terdapat batasan jumlah menu. Struktur hirarki dapat dilihat pada gambar 2-5.



Gambar 2. 5 Struktur Hirarki

4. Struktur Kombinasi

Struktur kombinasi adalah struktur navigasi yang mengkombinasikan struktur-struktur yang ada. Struktur kombinasi dapat dilihat pada gambar 2-6.



Gambar 2. 6 Struktur Kombinasi

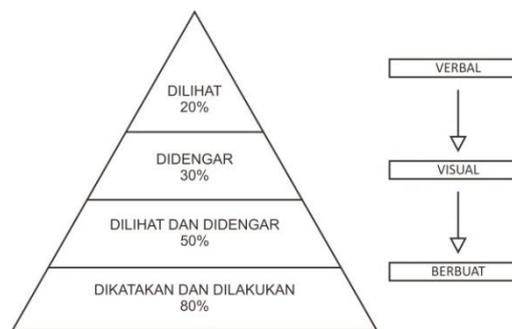
h. Aplikasi Pembelajaran Multimedia

Keberadaan teknologi informasi sangat berperan penting dalam meningkatkan kualitas hidup bagi pelajar sebagai awal proses interaksi terhadap perkembangan teknologi komputer, yang diharapkan nantinya dapat berguna pada kehidupan sehari-hari bagi pelajar.

Di antara berbagai aplikasi pembelajaran, multimedia memiliki suatu kelebihan tersendiri yang tidak dapat digantikan oleh media pembelajaran yang lain, dikarenakan pembelajaran multimedia memiliki unsur interaksi yang aktif, hiburan yang menyenangkan, serta sebagai sumber informasi yang lebih fleksibel untuk mendukung pelatihan terdistribusi pada kebutuhan pelajar. Unsur-unsur yang didapat dalam menggunakan aplikasi multimedia pembelajaran, meliputi :

1. Meningkatkan aliran gagasan serta informasi.
2. Merupakan cara yang kaya untuk mengomunikasikan sesuatu.
3. Mendorong partisipasi, keterlibatan dan eksplorasi pelajar.
4. Menstimulasi fungsi indera.
5. Kemudahan penggunaan, terutama bagi pelajar yang masih awam.

Dengan menggunakan aplikasi multimedia sebagai media pembelajaran, dapat meningkatkan proses penyerapan suatu informasi, dikarenakan manusia hanya mampu mengingat 20% dari apa yang dilihat, 30% dari yang didengar, 50% dari apa yang dilihat dan didengar, dan 80% dari apa yang didengar dan dilakukan secara bersamaan. Proses persentase penyerapan suatu informasi dapat dilihat pada gambar 2-7.



Gambar 2. 7 Kerucut Penyerapan Informasi

Pada tahap ini, dilakukan perumusan kebutuhan dan definisi umum dari aplikasi yang dikembangkan. Terdapat beberapa fungsi utama dalam tahap ini, antara lain :

1. Mendefinisikan aplikasi, meliputi makna dan tujuan dari aplikasi yang akan dibuat.
2. Mengevaluasi kebutuhan perangkat keras dan perangkat lunak, serta memilih kebutuhan perangkat yang paling baik untuk digunakan.

Memastikan kebutuhan pengguna untuk menggunakan aplikasi yang akan dibuat dan beberapa kebutuhan spesifik yang dijadikan sebagai bahan pertimbangan.

B. Sekilas Tentang Aplikasi Yang Digunakan

1. Web

Web merupakan media informasi berbasis jaringan komputer yang dapat diakses dimana saja dengan biaya relatif murah. *Web* merupakan bentuk implementasi dari bahasa pemrograman *Web* (*Web programming*). Sejarah perkembangan bahasa pemrograman *Web* diawali dengan munculnya HTML (*Hypertext Markup Language*), yang kemudian dikembangkan dengan munculnya CSS (*Cascading Style Sheet*) yang bertujuan untuk memperindah tampilan *Website*.

Bahasa pemrograman saat ini sudah sangat berkembang dengan berbagai kemudahan dalam penyajian dan *interface* yang lebih *friendly used*. Penyajian yang baik dari suatu Bahasa pemrograman akan menghasilkan sebuah *Web* yang dinamis sehingga pengunjung akan lebih mudah mendapatkan informasi yang dibutuhkan. Beberapa bahasa pemrograman *Web* berbasis *server* (*Server Side*) mulai muncul dan dikembangkan oleh beberapa perusahaan perangkat lunak, seperti : ASP (*Active Server Pages*) oleh *Microsoft*, JPS (*Java Server Pages*) oleh *Sun Microsystems*, dan CGI (*Common gateway Interface*) oleh perusahaan UNIX yang sekarang diadopsi oleh NCSA (*National Central For Supercomputing Application*) *Web server*. (Alexander F. K, Sibero. 2011)

2. Internet (*Interconneted Network*)

Internet adalah sekumpulan jaringan komputer yang saling terhubung secara fisik dan memiliki kemampuan untuk membaca dan menguraikan

protokol komunikasi tertentu yang disebut *Internet Protocol (IP)* dan *Transmission Control Protocol (TCP)*. Protocol adalah spesifikasi sederhana mengenai bagaimana komputer saling bertukar informasi. (Alan Nur Aditya, 2011)

3. WWW (*World Wide Web*)

World Wide Web atau WWW atau juga dikenal dengan *WEB* adalah suatu ruang informasi yang dipakai oleh pengenal global yang disebut pengidentifikasi sumber seragam untuk mengenal pasti sumber daya berguna. WWW sering dianggap sama dengan internet.

WWW merupakan kumpulan *Web* dari seluruh dunia yang mempunyai kegunaan untuk menyediakan data dan informasi untuk dapat digunakan bersama. WWW adalah bagian yang paling menarik dari internet. Melalui *Web*, para pengguna dapat mengakses informasi- informasi yang tidak hanya berupa teks tetapi bisa juga berupa gambar, suara, video dan animasi. WWW adalah suatu program yang ditemukan oleh Tim burners-Lee pada tahun 1991. Awalnya Berners Lee hanya ingin menemukan cara untuk menyusun arsip-arsip risetnya. Dengan program itu menciptakan jaringan yang manautkan berbagai arsip sehingga memudahkan pencairan informasi yang dibutuhkan. Inilah yang kelak menjadi dasar sebuah perkembangan pesat yang dikenal sebagai WWW.

4. HTTP (*Hypertext Tranfer Protocol*)

HTTP (*Hypertext Tranfer Protocol*) adalah protocol yang dipergunakan untuk mentransfer dokumen dalam *World Wide Web (WWW)*.

protocol ini adalah *protocol* ringan, tidak berstatus dan generik yang dapat dipergunakan berbagai macam tipe dokumen. Pengembangan HTTP dikoordinasikan oleh Konsorsium *World Wide Web* (W3C) dan grup bekerja *Internet Engineering Task Force* (IETF), bekerja dalam publikasi satu seri RFC, yang paling terkenal RFC yang menjelaskan HTTP/1,1, versi HTTP yang digunakan umum sekarang ini. HTTP adalah sebuah *protocol* meminta / menjawab antara *client* dan *server*.

Sebuah client HTTP seperti *Web browser*, biasanya melalui permintaan dengan membuat hubungan TCP/IP ke *port* tertentu. Sebuah *server* HTTP yang mendengarkan di *port* tersebut menunggu *client* mengirim kode permintaan di ikuti dengan pesan MIME yang memiliki beberapa informasi data tertentu. (Deka M Wildan, 2019).

5. URL (*Uniform Resource Locator*)

URL (*Uniform Resource Locator*) merupakan suatu *pathname* untuk mengidentifikasi sebuah dokumen di *Web*. Dibandingkan dengan *logical pathname* yang membuat nama direktori dan subdirektori yang menunjuk ke satu file di sebuah komputer, cakupan URL lebih luas lagi. Didalam URL dimuat lokasi atau nama mesin komputer yang diminta, nama *file* dari dokumen itu (termasuk *logical pathnamanya*), serta protoko untuk mengaksesnya. (Santoso, Harip. 2020)

6. HTML (*Hypertext Markup Language*)

HTML (*HyperText Markup Language*) Adalah sebuah bahasa markah yang digunakan untuk membuat sebuah halaman *Web*, menampilkan berbagai

informasi di dalam sebuah penjelajah *Web* internet dan penformatan hiperteks sederhana yang ditulis dalam berkas format ASCII agar dapat menghasilkan tampilan wujud yang terintegrasi. Dengan kata lain, berkas yang dibuat dalam perangkat lunak pengolah kata dan disimpan dalam format ASCII normal sehingga menjadi halaman *Web* dengan perintah-perintah HTML. Bermula dari sebuah bahasa yang sebelumnya banyak yang digunakan di dunia penerbitan dan pencetakan yang di sebut dengan SGML (*Standard Generalized Markup Language*), HTML adalah sebuah standar yang digunakan secara luas untuk menampilkan halaman *Web*. HTML saat ini merupakan standar internet yang didefinisikan dan dikendalikan penggunaannya oleh *World Wide Web Consortium* (W3C). HTML dibuat oleh kolaborasi Caillau TIM dengan Berners-Lee Robert ketika mereka bekerja di CERN pada tahun 1989 (CERN adalah lembaga penelitian fisika energi tinggi di Jenewa). (Santoso, Harip. 2010)

HTML juga bisa berupa dasar atau tulang punggung dalam pembuatan *Web*. Dalam setiap file HTML akan selalu di dahului dengan tag (HTML). Tag ini akan membuat *Web browser* mencoba untuk membaca file yang ada didalam tag tersebut untuk menampilkan dalam *Web browser*. (Santoso, Harip. 2019)

7. PHP (*Personal Home Page*)

PHP (*Personal Home Page*) merupakan bahasa pemrograman berbasis *Web* yang memiliki kemampuan untuk memproses dan mengolah data secara dinamis. PHP dapat dikatakan sebagai sebuah *server-side*

embedded script language, artinya semua sintaks dan perintah program yang ditulis akan sepenuhnya dijalankan oleh *server*, tetapi dapat disertakan pada halaman HTML biasa. Pada umumnya, semua aplikasi yang dibangun menggunakan PHP akan memberikan hasil pada *Web browser*, tetapi prosesnya secara keseluruhan dijalankan pada *server*. Pada prinsipnya, *server* akan bekerja apabila ada permintaan dari client. Dalam hal ini, client menggunakan kode-kode PHP akan mengirim perintah ke *server* ketika menggunakan PHP sebagai *server side embedded script language*. (Alan Nur Aditya, 2019)

8. MySQL

(Sutarman, 2020) MySQL merupakan salah satu program yang dapat digunakan sebagai database, dan merupakan salah satu *software* untuk database *server* yang banyak digunakan. MySQL bersifat *Open Source* dan menggunakan SQL. MySQL bisa dijalankan diberbagai platform misalnya *Windows, linux* dan lain sebagainya.

MySQL merupakan salah satu perangkat lunak sistem manajemen basis data (*database management system*) atau DBMS yang menggunakan perintah standar SQL (*Structured Query Language*). Dimana MySQL mampu untuk melakukan banyak eksekusi perintah *query* dalam suatu permintaan (*multithread*), baik itu mengirim dan menerima data. MySQL juga *multi-user* dalam arti dapat dipergunakan oleh banyak pengguna dalam waktu bersamaan. Dengan sekitar enam juta Instalasi diseluruh dunia. MySQL AB membuat MySQL tersedia dalam perangkat lunak gratis dibawah lisensi

GNU General Public License (GPL) dan juga menjual dalam lisensi komersial untuk keperluan jika penggunaanya tidak cocok menggunakan lisensi GPL. (Nugroho, Bunafit. 2019)

9. *PhpMyAdmin*

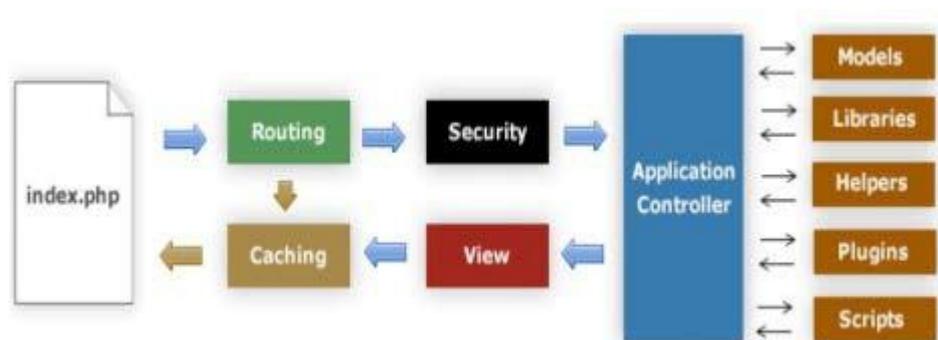
PhpMyAdmin adalah perangkat lunak yang ditulis dalam bahasa pemrograman *PHP* digunakan untuk menangani administrasi *MySQL* dengan menggunakan protokol *WWW* secara langsung dari *browser*. *PhpMyAdmin* mendukung berbagai operasi *MySQL*, diantaranya mengelola basis data, tabel-tabel, bidang *fields*, relasi *relations*, indeks, pengguna *users*, perijinan *permissions*, dan lain-lain. Dengan menggunakan *phpMyAdmin* diharapkan akan memudahkan dalam manajemen database *MySQL*. (Madcoms Madiun, 2020)

10. XAMPP (*Cross-Platform (x), Apache, MySQL dan Perl*)

XAMPP adalah perangkat lunak besar, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai *server* yang berdiri sendiri (*localhost*), yang terdiri atas program *Apache HTTP server*, *MySQL database* dan penerjemah bahasa yang ditulis dengan bahasa pemrograman *PHP* dan *Perl*. Nama XAMPP merupakan singkatan dari X (tempat sistem operasi apapun), *Apache*, *MySQL*, *PHP* dan *Perl*. Program ini tersedia dalam *GNU general Public license* dan bebas, merupakan *Web server* yang mudah digunakan yang dapat melayani tampilan halaman *Web* yang dinamis. (Madcoms Madiun, 2020)

11. *Framework Codeigniter*

Menurut (Madcoms Madiun, 2020) “*CodeIgniter* adalah sebuah *framework* PHP yang dapat membantu mempercepat *developer* dalam pengembangan aplikasi *Web* berbasis PHP dibanding jika menulis semua kode program dari awal”. Didalam CI ini terdapat beberapa macam kelas yang berbentuk *library* dan *helper* yang berfungsi untuk membantu pemrograman dalam mengembangkan aplikasi. File dokumentasi ini disertakan secara langsung pada saat Pengunduhan paket *framework CI*.



Gambar 2. 8 Codeigniter

C. Perancangan Basis Data

Menurut (Jogiyanto Hartono, 2021). Basis data (*Database*) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di simpan luar komputer dan digunakan perangkat lunak tertentu untuk memanipulasi. Sedangkan menurut penadapat saya Basis Data adalah data data yang tersimpan dalam komputer dan masih harus di proses agar saling berhubungan satu sama lain. *Database* merupakan salah satu komponen yang

penting di sistem informasi, karena berfungsi sebagai basis penyedia informasi bagi para pemakai. Sistem basis data ini adalah suatu sistem informasi yang mengintegrasikan kumpulan dari data yang saling berhubungan satu dengan lainnya dan membuatnya tersedia untuk beberapa aplikasi yang bermacam-macam di dalam suatu organisasi.

1. Entity Relationship Diagram (ERD)

Menurut Andi, ERD berguna untuk mengorganisasi data-data (*hanya data*) dalam hubungannya dengan data-data yang lain. Sedangkan menurut saya ERD berfungsi untuk mengelompokkan data- data agar terjadi keselarasan hubungan antar data lainnya. ERD mempunyai koleksi objek-objek dasar yang dinamakan entitas serta hubungan (*relationship*) antara entitas –entitas itu.

Entitas adalah sesuatu atau objek pada dunia nyata yang dapat dibedakan satu terhadap yang lainnya, yang bermanfaat bagi aplikasi yang sedang kita kembangkan. Sedangkan *Relationship* adalah hubungan antara beberapa entitas.

2. Normalisasi

Menurut Andi, Normalisasi sebagai proses untuk mengubah suatu relasi yang memiliki masalah tertentu ke dalam dua buah relasi atau lebih yang tak memiliki masalah tersebut. Sedangkan Normalisasi menurut pendapat saya adalah proses pengolahan data yang masih salah dan memiliki data ganda untuk kemudian di perbaiki sehingga menghasilkan relasi-relasi yang lebih baik dari sebelumnya. Normalisasi dapat dipahami sebagai tahapan-tahapan yang masing- masing berhubungan dengan bentuk normal. Bentuk normal

adalah keadaan relasi yang dihasilkan dengan menerapkan aturan sederhana yang berkaitan dengan konsep kebergantungan fungsional pada relasi yang bersangkutan sebagai berikut.

a. Bentuk normal pertama.

Bentuk normal pertama adalah suatu bentuk relasi dimana atribut bernilai banyak (*Multivalued attribute*) telah dihilangkan sehingga kita akan menjumpai nilai tunggal (*mungkin saja nilai null*) pada perpotongan setiap baris dan kolom.

b. Bentuk normal kedua.

Suatu relasi adalah dalam bentuk normal kedua jika dia berada dalam bentuk normal pertama dan setiap atribut bukan kunci bergantung penuh pada kunci primer.

c. Bentuk normal ketiga.

Relasi adalah dalam bentuk normal ke tiga adalah jika berada dalam bentuk normal kedua dan tidak dijumpai kebergantungan *transitif*. Kebergantungan *transitif* dalam suatu relasi adalah kebergantungan fungsional antara 2 atau lebih atribut bukan kunci. Ketika relasi memiliki lebih dari 1 kunci kandidat, anomali-anomali seperti kita bahas di atas masih mungkin dijumpai meskipun relasi yang bersangkutan sudah dalam bentuk normal ke tiga.

3. Tabel Relasi

Tabel relasi adalah tabel hubungan antara suatu himpunan entitas dengan entitas yang lainnya.

4. Kunci

Kunci (*key*) merupakan suatu atribut yang unik yang dapat digunakan untuk membedakan suatu entitas dengan entitas yang lainnya dalam suatu himpunan entitas.

D. Tahap-tahap Perancangan Sistem

1. Use Case

(Sukamto, Rosa Ariani. 2019) *Use Case diagram* adalah abstraksi dari interaksi antara sistem dan *actor*. *Use Case* bekerja mendeskripsikan tipe interaksi antara *user* sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai.

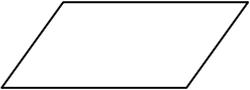
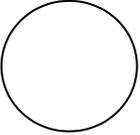
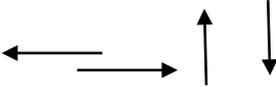
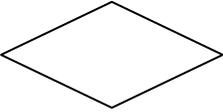
2. Kamus Data

Kamus data (*data dictionary*) adalah katalog fakta tentang data dan kebutuhan informasi yang mengalir pada sistem dengan lengkap dari suatu sistem informasi. Kamus data di buat berdasarkan arus data yang mengalir pada *Data Flow Diagram* (DFD). Kamus data merupakan tempat penyimpanan dari aliran-aliran data, *file-file* dan proses-proses dalam sebuah sistem. Bagian ini menjelaskan secara detail proses-proses yang terjadi disetiap proses, *file* dan struktur data mengenai model sistem yang digambarkan. Kamus data digunakan untuk menjelaskan semua data yang mengalir atau digunakan dalam sistem, yaitu mengenai arus data yang masuk ke dalam sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem.

3. Bagan Alir Program

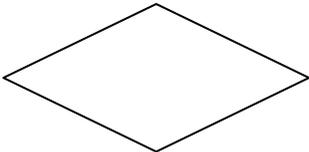
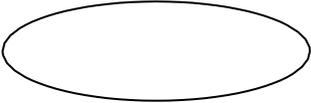
Bagan alir program (*program flowchart*) merupakan bagan yang menjelaskan rinci langkah-langkah dari proses program. Bagan alir program dibuat dari derifikasi bagan alir sistem (Bagan alir program dibuat dengan menggunakan symbol-simbol sebagai berikut :

Tabel 2. 1 Simbol *Flowchart*

Simbol	Keterangan
	Simbol <i>input/output</i> digunakan untuk mewakili data <i>input/output</i>
	Simbol proses, digunakan untuk mewakili suatu proses
	Simbol proses terdefinisi, digunakan untuk menunjukkan suatu operasi yang rinciannya ditunjukkan ditempat lain
	Simbol titik terminal, digunakan untuk awal dan akhir suatu proses
	Simbol penghubung, digunakan untuk menunjukkan sambungan dari alir yang terputus dihalaman yang masih sama
	Simbol garis alir, menunjukkan arus dari proses
	Simbol keputusan, digunakan untuk suatu penyelesaian kondisi didalam program

4. Entity Relationship Diagram

Tabel 2. 2 Simbol Entity Relationship Diagram

NOTASI	KETERANGAN
	Entitas, yaitu kumpulan dari objek yang diidentifikasi secara unik
	Relasi, yaitu hubungan yang terjadi antara satu atau lebih entitas. Jenis hubungan antara lain : satu ke satu, satu ke banyak dan banyak ke banyak
	Atribut, yaitu karakteristik dari entity atau relasi yang merupakan penjelasan detail tentang entitas.
	Garis, hubungan antara entity dengan atributnya dan himpunan entitas dengan himpunan relasi.
	<i>Input/Output</i> data, yaitu proses <i>input/output</i> data, parameter, informasi.

Entity Relationship Diagram (ERD) adalah adalah suatu pemodelan konseptual yang didesain secara khusus untuk mengidentifikasi entitas yang menjelaskan data dan hubungan antar data. ERD merupakan suatu model untuk menjelaskan hubungan antardata dalam basis data berdasarkan objek-objek dasar data yang mempunyai hubungan antar relasi. ERD untuk

memodelkan struktur data dan hubungan antardata, untuk menggambarannya digunakan beberapa notasi dan simbol. Pada dasarnya ada tiga komponen yang digunakan, yaitu :

a. Entity

Entity merupakan objek yang mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Simbol dari entiti ini biasanya digambarkan dengan persegi panjang.

b. Atribut

Setiap entitas pasti mempunyai elemen yang disebut atribut yang berfungsi untuk mendeskripsikan karakteristik dari entitas tersebut. Isi dari atribut mempunyai sesuatu yang dapat mengidentifikasi isi elemen satu dengan yang lain. Gambar atribut diwakili oleh simbol *elips*.

5. UML (*Unified Modelling Language*)

UML (*Unified Modelling Language*) menurut menurut Adi nugroho (2019), “*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) *Behaviorial 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 memperjelas 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):

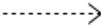
- 1) Ketergantungan (*dependency*) adalah hubungan dimana perubahan yang terjadi pada suatu elemn independent akan mempengaruhi elemen yang bergantung padanya.
- 2) Asosiasi adalah apa dan bagaimana yang menghubungkan antara objek satu dengan yang lainnya. Suatu bentuk asosiasi adalah agregasi yang menampilkan hubungan suatu objek dengan bagian-bagiannya.

- 3) Generalisasi adalah hubungan dimana objek anak berbagai perilaku dan struktur data dari objek yang ada di atasnya (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).
- d. 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 pemakai

Adapun simbol-simbol *Use Case* Diagram antara lain :

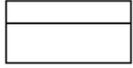
Tabel 2. 3 Simbol *Use Case* Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>Use Case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).

NO	GAMBAR	NAMA	KETERANGAN
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>Use Case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>Use Case</i> target memperluas perilaku dari <i>Use Case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (<i>sinergi</i>).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

Tabel 2. 4 Simbol *Class Diagram*

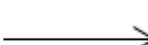
NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.

NO	GAMBAR	NAMA	KETERANGAN
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya

Tabel 2. 5 Simbol *Sequence Diagram*

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

Tabel 2. 6 Simbol *StateChart* Diagram

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

Tabel 2. 7 Simbol *Activity* Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

E. Tumbuh Kembang Anak 0-5 Tahun

Masa bayi adalah masa keemasan sekaligus masa kritis perkembangan seseorang. Dikatakan masa kritis karena pada masa ini bayi sangat peka terhadap lingkungan dan dikatakan masa keemasan karena masa bayi berlangsung sangat singkat dan tidak dapat diulang kembali (Departemen Kesehatan, 2019). Usia perkembangan bayi terbagi 2 yaitu, neonatus sejak lahir sampai usia 28 hari dan bayi dari usia 29 hari sampai 12 bulan (WHO, 2019)

sedangkan untuk balita 1 tahun perempuan mempunyai berat badan mulai dari 7-13,2 kg. Sedangkan untuk berat badan balita 2 tahun (13-24 bulan) mempunyai berat untuk anak laki-laki adalah 9,7-15,3 kg untuk anak berusia 2 tahun dan 10,5-16,9 kg untuk anak laki-laki berusia 2 tahun 6 bulan.

Usia 1 tahun: 70–78 sentimeter. Usia 2 tahun: 80–92 sentimeter. Usia 3 tahun: 82–95 sentimeter. Usia 4 tahun: 83–96 sentimeter. (Merita, 2019)

F. Penelitian Terdahulu

Berbagai penelitian sebelumnya merupakan hal yang sangat perlu dan dapat dijadikan sebagai data pendukung. Salah satu data pendukung yang menurut peneliti perlu dijadikan bagian tersendiri adalah penelitian terdahulu yang relevan dengan permasalahan yang sedang dibahas dalam penelitian ini. Dalam hal ini, fokus penelitian terdahulu yang dijadikan acuan adalah terkait dengan masalah teknologi informasi. Oleh karena itu, peneliti melakukan kajian terhadap beberapa hasil penelitian berupa tugas akhir/skripsi atau jurnal melalui internet.

1. Merita. (2019) program studi Program Studi Ilmu Gizi, STIKes Baiturrahim Jambi, Melakukan Penelitian Yang Berjudul “TUMBUH KEMBANG ANAK

USIA 0-5 TAHUN” Beberapa dampak akibat dari tumbuh kembang anak yang tidak sesuai dengan usianya diantara lain dapat menghambat perkembangan otak. Oleh karena itu, tujuan pengabdian ini adalah untuk melakukan penyuluhan tumbuh kembang pada anak balita. Kegiatan ini dilaksanakan pada bulan September 2018 sampai Januari 2019, di Posyandu Sukasari Kabupaten Sarolangun. Sasaran adalah ibu yang mempunyai anak usia 0-5 tahun. Kegiatan ini terdiri dari pemberian penyuluhan dengan media power point, leaflet, poster, serta soal pre-post test dan pengukuran antropometri balita. Luaran dari kegiatan pengabdian ini yaitu: (1) Media leaflet dan poster tumbuh kembang anak dan; (2) Perilaku ibu dalam pemantauan tumbuh kembang anak. Hasil kegiatan menunjukkan bahwa peserta mampu menjawab pertanyaan terkait dengan materi tumbuh kembang pada anak usia 0-5 tahun pada tahap post-test, yaitu jawaban benar sebesar $\geq 80\%$. Peserta juga mampu memahami materi tentang tumbuh kembang anak terkait dengan pola konsumsi yang baik melalui diskusi. Berdasarkan kegiatan ini maka perlu upaya kerja sama dari petugas kesehatan dan kader untuk memotivasi ibu balita agar rutin melakukan pemantauan tumbuh kembang balita ke posyandu atau fasilitas pelayanan kesehatan terdekat.

2. HELMI LUTFIAN (2021), Program Studi Teknik Informatika Fakultas Komunikasi Dan Informatika Universitas Muhammadiyah Surakarta 2022, dalam penelitiannya yang berjudul “**SISTEM INFORMASI POSYANDU BERBASIS WEB**” Posyandu telah lama dikenal sebagai pusat pelayanan kesehatan primer bagi ibu dan anak, saat ini sangat dibutuhkan untuk

mempercepat penurunan angka kematian ibu dan anak dan dukungan masyarakat dan menyelenggarakan kegiatan pelayanan Kesehatan dasar. Berdasarkan pada hasil survey awal menggambarkan bahwa masih banyak paraiibu rumah tangga produktif (usia subur) masih kurang memahami mengenai kesehatan Ibu dan anak karena kurangnya informasi yang mereka dapatkan. Informasi ini berguna untuk mengetahui pertumbuhan anak, apakah ada penyimpangan atau tidak. Jika Terjadi penyimpangan segera diketahui dan ditindaklanjuti oleh tenaga medis di Puskesmas. Berdasarkan permasalahan yang dihadapi serta didukung penelitian sebelumnya maka penelitian untuk mengembangkan sistem informasi Posyandu berbasis Web. Sistem ini memiliki fitur pencatatan penimbangan dan imunisasi sekaligus mencetak laporan posyandu, dan mengolah permintaan laporan. Metode yang digunakan dalam penelitian ini yaitu waterfall terdiri dari analisis kebutuhan, desain, penulisan kode program, pengujian, penerapan. Sistem ini dibuat menggunakan Bahasa pemrograman PHP dan Framework Codeigniter. Pengujian Menggunakan System Usability Scale (SUS) dan Blackbox. Hasil pengujian blackbox testing menunjukkan 100% bahwa sistem sudah berjalan dengan baik dan setiap tombol sudah berjalan sesuai dengan fungsinya tanpa ada eror. Pada pengujian SUS dihasilkan nilai 80, sehingga sistem dinilai bekerja dengan baik dan layak untuk digunakan dalam melakukan pelaporan posyandu dengan baik..

3. Dio Arisyahputra (2020), Program Studi Sistem Komputer, Fakultas Teknik dan Ilmu Komputer, **Universitas** Komputer Indonesia dalam penelitiannya

yang berjudul **“Perancangan Aplikasi Pemantauan Tumbuh Kembang Anak Berbasis Android”** ada penelitian ini dibangun aplikasi pemantauan tumbuh kembang anak berbasis android. Pada penelitian ini untuk mengetahui pertumbuhan anak adalah dengan menggunakan standar antropometri, antropometri sendiri merupakan standar pengukuran pertumbuhan dengan menghitung berat badan, panjang atau tinggi badan, dan lingkaran kepala. Sedangkan untuk mengetahui perkembangan anak, orang tua dapat menggunakan KPSP (Kuisisioner Praskrining Perkembangan) dan menjawab pertanyaan yang ada berdasarkan kemampuan yang dialami oleh anak. Hasil dari KPSP sendiri adalah dengan menghitung jumlah jawaban “Ya”. Dengan dibangunnya system ini diharapkan orang tua dapat mengetahui dan memantau tumbuh kembang anak dengan mudah melalui Android.

BAB III

METODOLOGI PENELITIAN

A. Waktu Dan Lokasi Penelitian

Pelaksanaan penelitian dilakukan pada bulan Mei 2023 dan lokasi Puskesmas dan posiyandu Kota Parepare.

B. Jenis Penelitian

Penulis dalam hal ini menggunakan metode penelitian Deskriptif yaitu kegiatan yang dilakukan dengan cara mengumpulkan data secara langsung dari objek penelitian melalui wawancara, pengamatan langsung dan pengumpulan dokumen.

C. Metode Pengumpulan Data

Di penelitian ini, penulis memperoleh data dengan metode sebagai berikut:

1. Observasi

Dilakukan dengan mengadakan penelitian langsung dengan kantor yang terkait untuk mengumpulkan informasi dan data yang dibutuhkan dalam menunjang permasalahan.

2. Wawancara

Wawancara adalah bentuk komunikasi langsung antara peneliti dengan responden. Komunikasi ini berlangsung dalam bentuk tanya jawab dalam hubungan tata muka guna memperoleh data langsung

3. Studi Pustaka

Studi pustaka adalah suatu metode pengumpulan data dengan menggunakan buku-buku, laporan-laporan atau bacaan lain sebagai bahan referensi dalam penulisan laporan dan pembuatan sistem.

D. Alat Dan Bahan Penelitian

Dalam melakukan penelitian, penulis memerlukan alat dan bahan penelitian yang mendukung kegiatan tersebut. Alat dan bahan yang diperlukan antara lain:

1. Perangkat keras

Perangkat keras yang digunakan dalam pembuatan Aplikasi Multimedia Interaktif Untuk Mengetahui Tumbuh Kembang Anak Berdasarkan Pengelolaan Posyandu Berbasis Web adalah sebuah laptop dengan spesifikasi minimal yang dapat dilihat pada tabel 3.1 berikut :

Tabel 3. 1 Spesifikasi Minimal Perangkat Keras

Jenis	Spesifikasi
Laptop	Asus
Processor	Core-i3-3217U 1,8 GHz
Memory	4 GB
Hardisk	500 GB

2. Perangkat Lunak

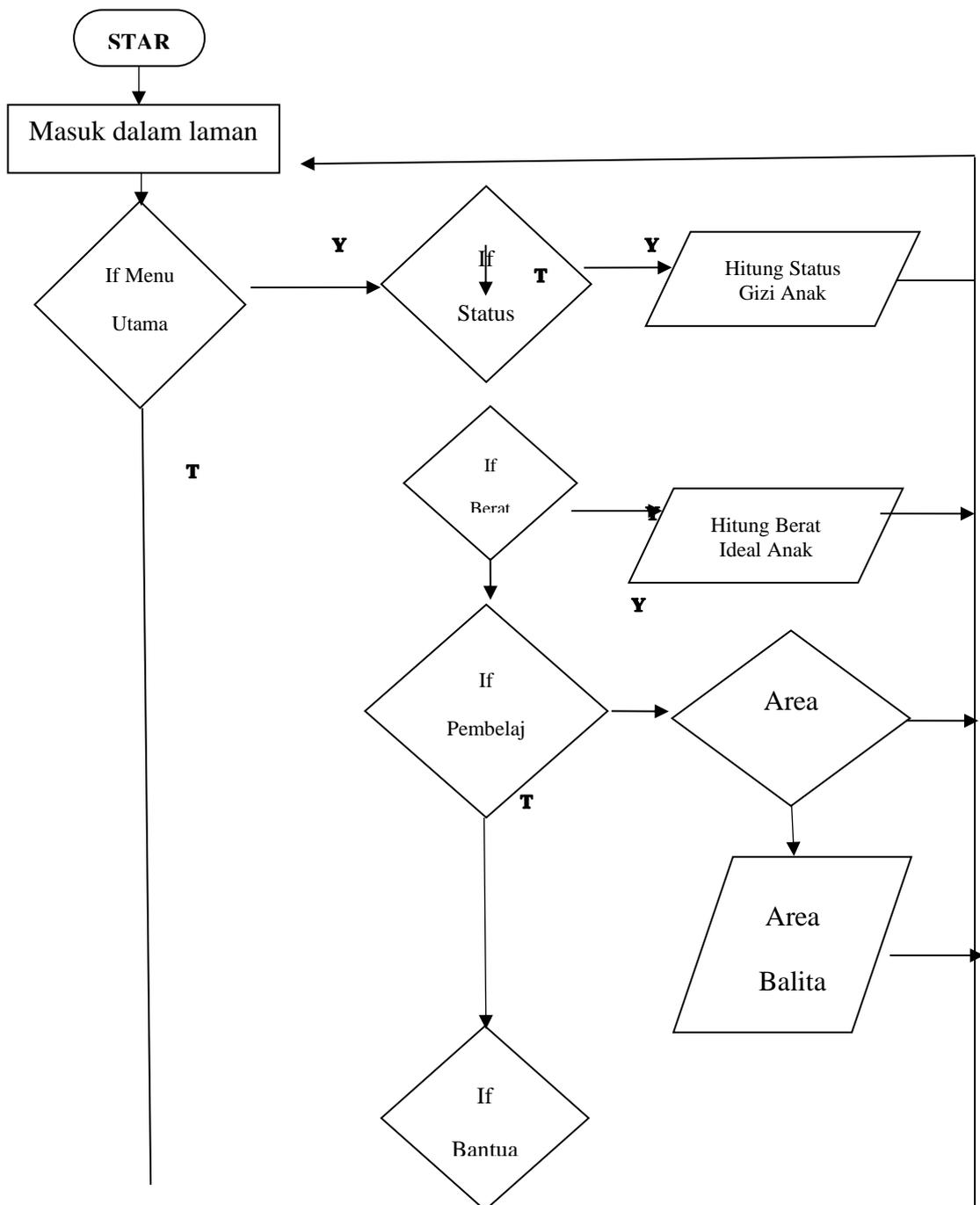
Perangkat lunak yang digunakan untuk membangun aplikasi ini dapat dilihat pada

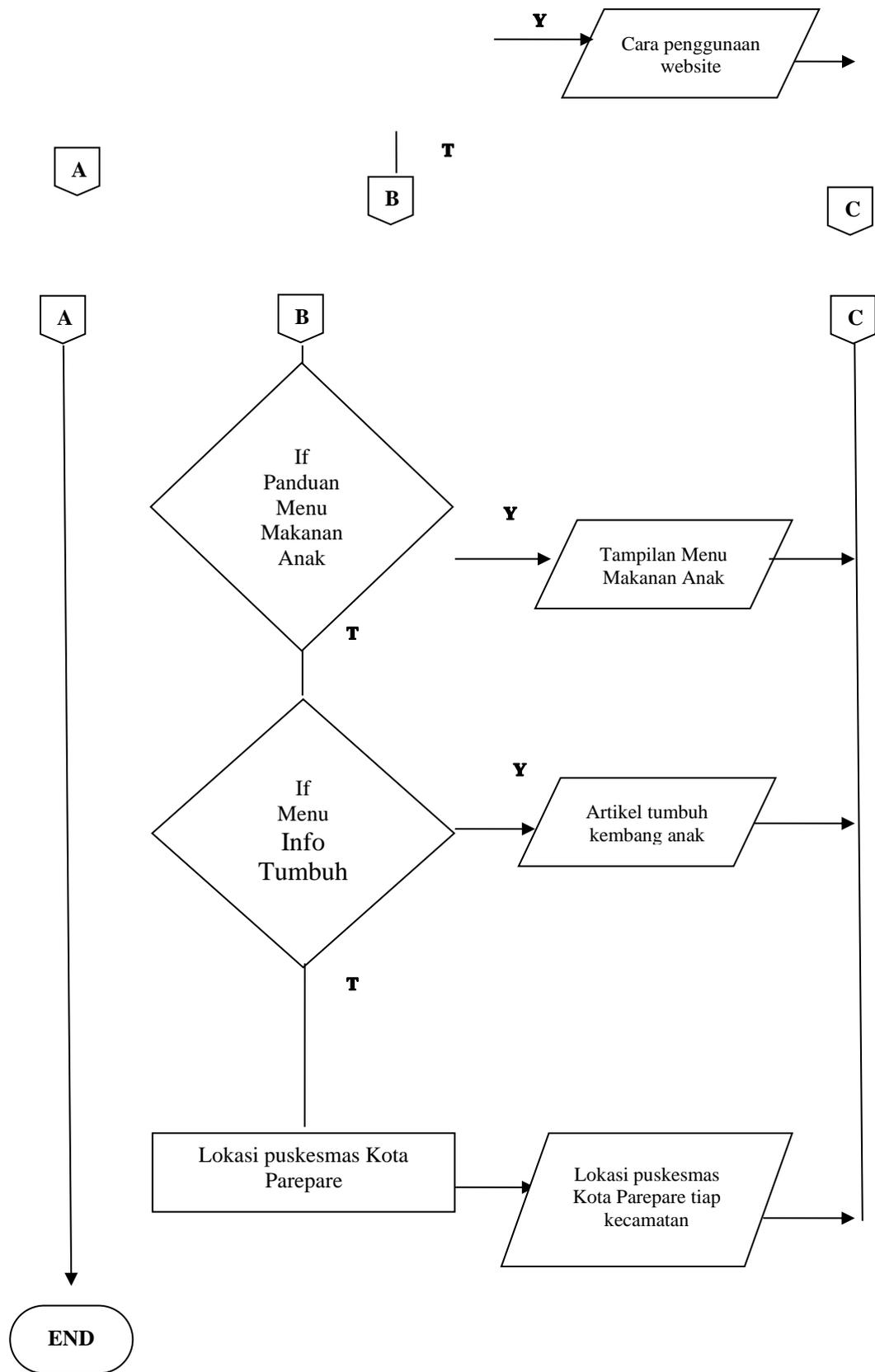
Tabel 3.2 berikut :

Tabel 3. 2 Spesifikasi perangkat Lunak

Jenis	Spesifikasi
Sistem Operasi	Windows 11
Bahasa Pemrograman	PHP
Database	MySql
Tools	Vscode,laragon

E. Desain Sistem





Gambar 3. 1 Diagram Alir Aplikasi

Pada diagram alir aplikasi di atas dapat dijelaskan sebagai berikut :

1. Mulai dari *start* diteruskan ke halaman utama.
2. Pada menu utama dengan berisikan 7 menu yaitu menu cek status gizi , menu cek berat ideal , menu pembelajaran anak , menu bantuan, menu resep, menu info tumbuh kembang anak dan menu lokasi puskesmas kota parepare.
3. Apabila memilih menu cek status gizi maka akan menampilkan form pengisian untuk di isi user dalam penghitungan status gizi anak.
4. Apabila memilih menu cek status berat badan ideal anak maka akan menampilkan form pengisian untuk di isi user dalam penghitungan status berat badan ideal anak.
5. Apabila memilih pembelajaran anak maka akan diarahkan ke 2 tombol navigasi untuk memilih media pembelajaran anak batita atau balita.
6. Apabila memilih bantuan maka akan ditampilkan petunjuk penggunaan website secara detail.
7. Apabila memilih menu resep maka akan ditampilkan resep – resep makanan untuk anak – anak
8. Pada menu Informasi tumbuh kembang anak akan ditampilkan artikel informasi seputar pertumbuhan anak.
9. Jika memilih menu lokasi puskesmas Parepare maka akan ditunjukkan peta Kota Parepare yang berisikan lokasi puskesmas pada setiap kecamatan

BAB IV

HASIL DAN PEMBAHASAN

A. Analisis Aliran Data UML

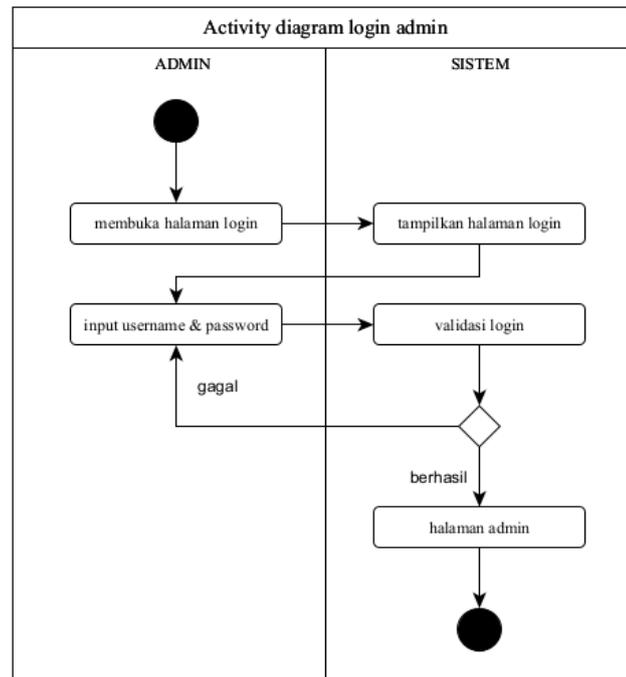
Tujuan dalam perencanaan sistem ini adalah untuk lebih mendalam tentang proses pembelajaran, khususnya terkait pertumbuhan dan perkembangan anak, baik dalam aspek gizi maupun berat badan mereka. Pada saat ini, peneliti sedang merancang pengembangan sistem yang berfokus pada objek, dengan menggunakan diagram use case, diagram aktivitas, dan diagram urutan.

1. Activity Diagram

Activity diagram adalah bentuk standar bahasa yang digunakan untuk mendokumentasikan, menentukan, dan mengembangkan perangkat lunak. Diagram aktivitas ini menggambarkan aktivitas-aktivitas yang terjadi dalam alur proses dalam suatu sistem.

a. Activity Diagram *Admin*

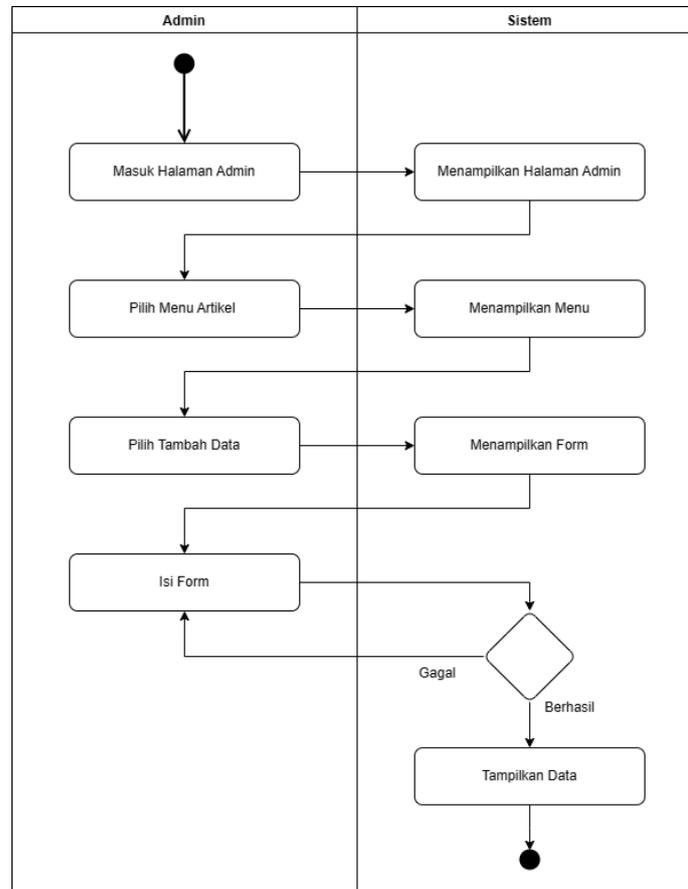
1) Activity diagram login



Gambar 4. 1 *Activity Diagram Login*

Pada Gambar 4.1 Menguraikan prosedur login yang dilakukan oleh seorang *Admin*. Langkah awal adalah *Admin* membuka situs *website*, memilih opsi "*login*" di menu, setelah itu sistem akan menampilkan formulir *login*. Kemudian, *Admin* diminta untuk memasukkan *Email* dan *Password*. Setelah itu, sistem akan memeriksa keabsahan informasi yang dimasukkan. Jika informasi tersebut benar, *Admin* akan diarahkan ke halaman dashboard. Namun, jika ada kesalahan, *Admin* akan kembali ke halaman login dan akan diberikan pesan kesalahan.

2) Activity diagram Tambah Data

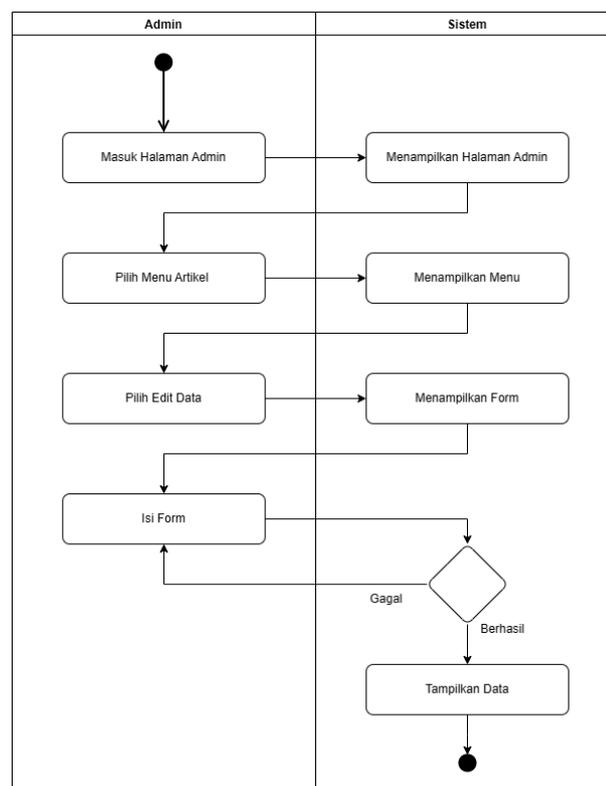


Gambar 4. 2 Activity Diagram Tambah Data

Pada gambar 4.2 Menjelaskan langkah-langkah yang harus diikuti oleh seorang *Admin* untuk menambahkan data. Awalnya, *Admin* perlu mengakses halaman *Admin*, yang kemudian akan memunculkan *dashboard* sistem. Selanjutnya, *Admin* harus memilih menu di mana data akan ditambahkan, sehingga sistem akan menampilkan halaman menu yang terkait. *Admin* selanjutnya harus memilih opsi "tambah data," sehingga sistem akan menampilkan formulir untuk menambahkan data. Setelah mengisi formulir, sistem akan melakukan validasi data. Jika validasi berhasil, sistem akan menampilkan halaman yang berisi data yang baru

ditambahkan. Sebaliknya, jika validasi gagal, sistem akan memberikan pesan kesalahan, dan *Admin* harus mengisi ulang formulir tambah data dengan benar.

3) Activity diagram Ubah Data

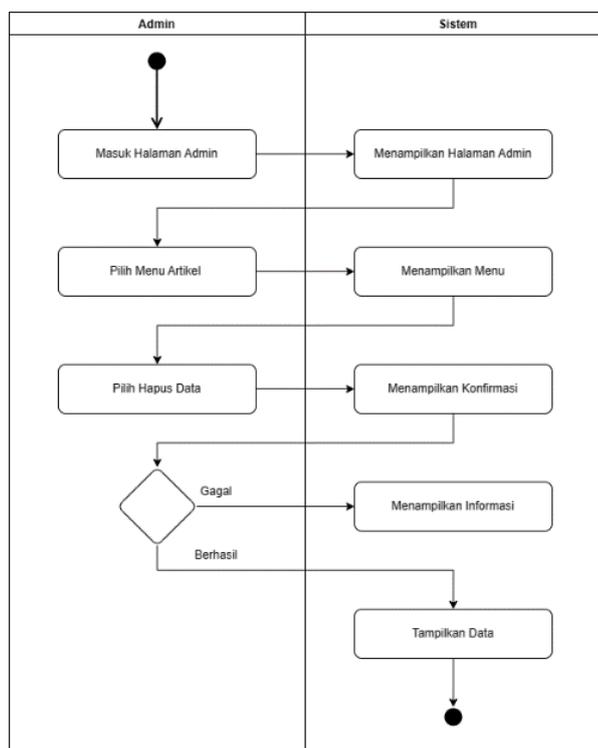


Gambar 4. 3 Activity Diagram Ubah Data

pada gambar 4.3 Menjelaskan langkah-langkah yang perlu diambil oleh seorang *Admin* untuk memodifikasi artikel. Awalnya, *Admin* perlu mengakses halaman artikel, yang akan mengarahkannya ke tampilan kelola artikel. Selanjutnya, *Admin* harus memilih menu yang berisi data yang ingin diubah, dan ini akan membuka halaman menu yang dipilih oleh *Admin*. Selanjutnya, *Admin* perlu memilih opsi "ubah data," yang akan membawa *Admin* ke formulir untuk mengubah data. Setelah mengisi formulir ubah data, sistem akan melakukan validasi data. Jika validasi berhasil, sistem akan menampilkan halaman yang berisi data yang telah diubah. Namun, jika validasi gagal, sistem akan memberikan pesan

kesalahan, dan *Admin* diharapkan mengisi kembali formulir ubah data dengan benar.

4) Activity diagram Hapus Data



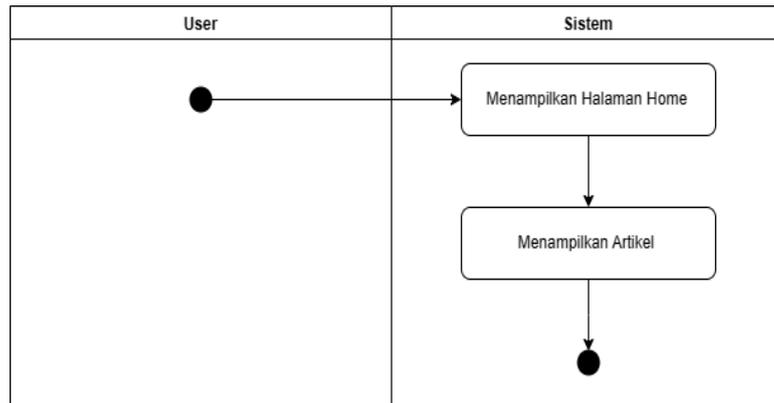
Gambar 4. 4 Activity Diagram Hapus Data

pada gambar 4.4 Menjelaskan langkah-langkah yang diambil oleh seorang *Admin* untuk menghapus data. Pertama-tama, *Admin* perlu membuka halaman artikel, yang akan mengarahkannya ke tampilan kelola artikel. Selanjutnya, *Admin* harus memilih menu yang berisi data yang ingin dihapus, dan ini akan membuka halaman menu yang dipilih oleh *Admin*. Setelah itu, *Admin* perlu memilih opsi "hapus data," yang akan memunculkan konfirmasi penghapusan. Setelah muncul konfirmasi, *Admin* harus melakukan validasi. Jika *Admin* memilih "ya," maka sistem akan menghapus data tersebut dan menampilkan halaman menu yang berisi data yang telah dihapus oleh *Admin*. Sebaliknya, jika *Admin* memilih "tidak," sistem

tidak akan menghapus data tersebut dan langsung menampilkan halaman data tersebut.

b. Activity Diagram *User*

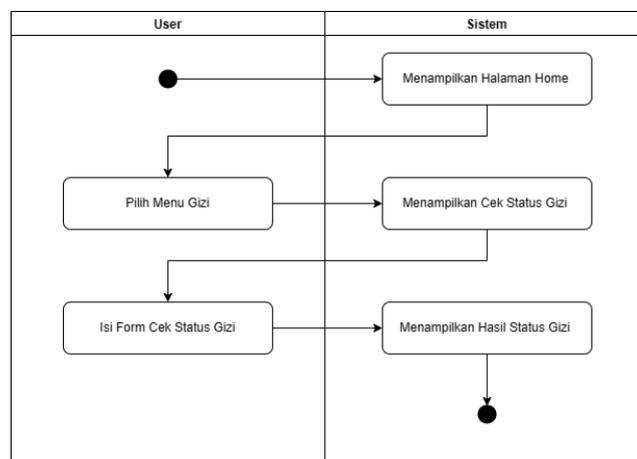
1) Activity Diagram pilih menu Artikel



Gambar 4. 5 *Activity* Diagram lihat Artikel

pada gambar 4.5 Menjelaskan langkah-langkah yang diambil oleh seorang pengguna untuk menampilkan halaman Artikel. Awalnya, pengguna harus membuka situs web, yang akan menghasilkan tampilan halaman utama. Selanjutnya, pengguna diminta untuk memilih menu "Artikel," yang akan membawa mereka ke bagian Artikel dalam sistem.

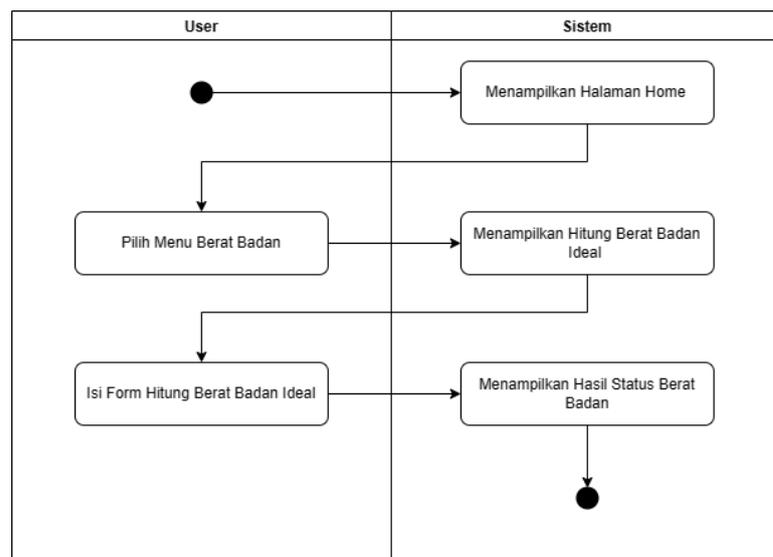
2) Activity Diagram cek Gizi



Gambar 4. 6 *Activity* Diagram Cek Gizi

pada gambar 4.6 Menjelaskan langkah-langkah yang diambil oleh seorang pengguna untuk menampilkan halaman Cek Gizi. Pertama-tama, pengguna harus membuka situs web, yang akan mengarahkan mereka ke halaman utama. Selanjutnya, pengguna diharapkan untuk memilih opsi "Gizi" dari menu, yang akan membawa mereka ke bagian formulir Cek Gizi dalam sistem. Pengguna diminta untuk mengisi formulir sesuai dengan instruksi yang tertera pada formulir. Setelah formulir diisi, sistem akan menampilkan hasil perhitungan gizi sesuai dengan kriteria yang telah dimasukkan oleh pengguna.

3) Activity Diagram cek Berat Badan

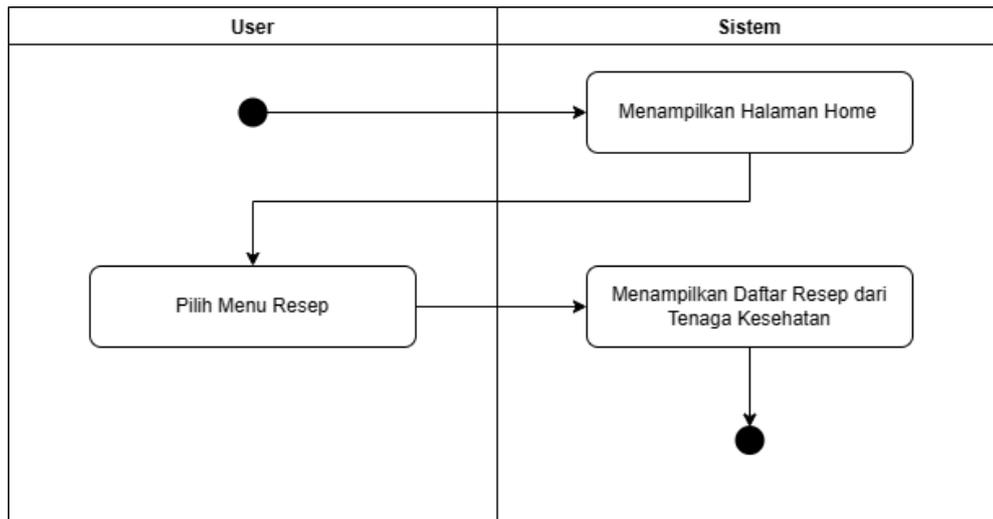


Gambar 4. 7 Activity Diagram Cek Berat Badan

pada gambar 4.7 Menjelaskan langkah-langkah yang diambil oleh seorang pengguna untuk memeriksa berat badan ideal. Awalnya, pengguna harus membuka situs web, yang akan mengarahkannya ke halaman utama. Selanjutnya, pengguna diharapkan untuk memilih opsi "berat badan" dari menu, yang akan membawa mereka ke bagian formulir pemeriksaan berat badan dalam sistem. Pengguna

diharapkan untuk mengisi formulir sesuai dengan petunjuk yang ada pada formulir tersebut. Setelah mengisi formulir, hasil perhitungan berat badan ideal sesuai dengan kriteria yang telah dimasukkan oleh pengguna akan ditampilkan.

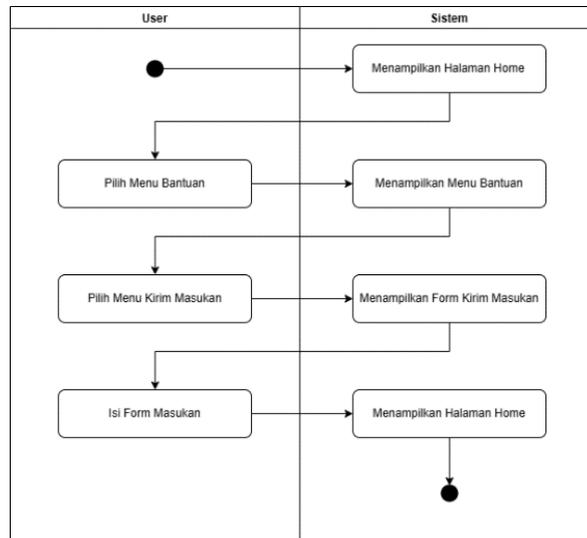
4) Activity Diagram menampilkan resep



Gambar 4. 8 *Activity Diagram Menampilkan Resep*

pada gambar 4.8 Menjelaskan langkah-langkah yang diambil oleh seorang pengguna untuk menampilkan resep. Pertama-tama, pengguna harus membuka situs web, yang akan mengarahkannya ke halaman utama. Selanjutnya, pengguna diharapkan untuk memilih opsi "resep" dalam menu navigasi yang tersedia. Setelah itu, sistem akan menampilkan resep yang telah disediakan oleh dokter dan petugas kesehatan.

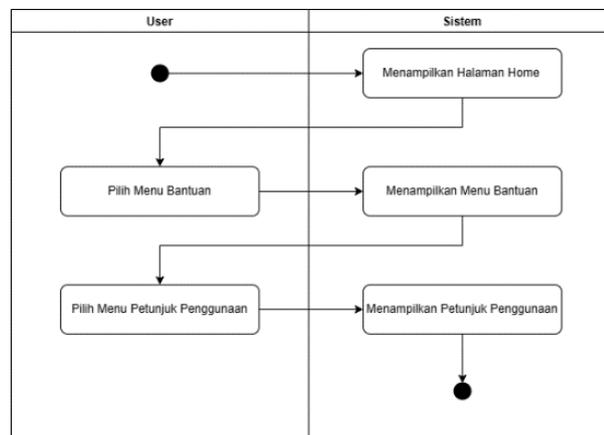
5) Activity Diagram kirim masukan



Gambar 4. 9Activity Diagram Kirim Masukan

pada gambar 4.9 Menjelaskan langkah-langkah yang diambil oleh seorang pengguna untuk mengirimkan masukan. Pertama-tama, pengguna harus membuka situs web, yang akan mengarahkannya ke halaman utama. Selanjutnya, pengguna diharapkan untuk memilih opsi "bantuan" dan kemudian memilih pilihan "kirim masukan" dari menu. Setelah itu, sistem akan menampilkan formulir kirim masukan yang harus diisi oleh pengguna.

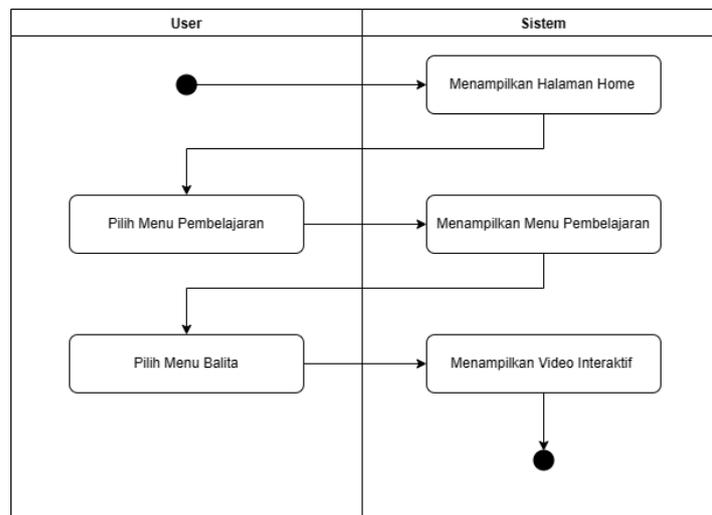
6) Activity Diagram petunjuk penggunaan aplikasi



Gambar 4. 10Activity Diagram Petunjuk Penggunaan Aplikasi

pada gambar 4.10 Menjelaskan langkah-langkah yang diambil oleh seorang pengguna untuk melihat panduan penggunaan aplikasi. Pertama-tama, pengguna perlu membuka situs web, yang akan mengarahkannya ke halaman utama. Selanjutnya, pengguna diminta untuk memilih opsi "bantuan" dan kemudian memilih opsi "petunjuk penggunaan" dari menu. Setelah itu, sistem akan menampilkan panduan penggunaan aplikasi yang mencakup semua fitur yang tersedia.

7) Activity Diagram menampilkan pembelajaran balita

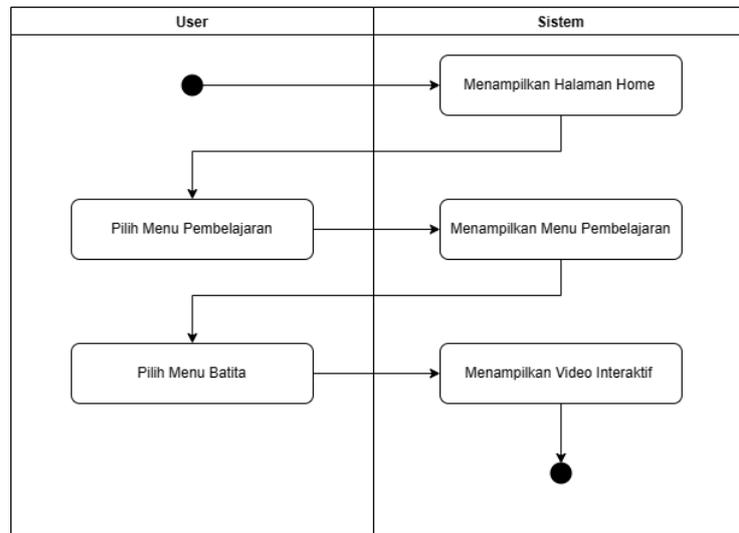


Gambar 4. 11 *Activity* Diagram Pembelajaran Balita

Pada gambar 4.11 Menggambarkan langkah-langkah yang harus dilakukan oleh pengguna untuk mengakses materi pembelajaran tentang anak balita. Awalnya, pengguna perlu membuka situs web, yang selanjutnya akan mengarahkan mereka ke halaman utama. Setelah itu, pengguna akan memilih opsi "Pembelajaran" dari menu navigasi yang tersedia. Dalam menu tersebut, pengguna akan diberikan pilihan antara materi pembelajaran untuk balita atau batita. Setelah memilih salah satu dari opsi tersebut, pengguna kemudian diminta untuk memilih materi

pembelajaran yang ingin diakses, dan pada saat itu, sistem akan menampilkan video interaktif sesuai dengan pilihan pengguna.

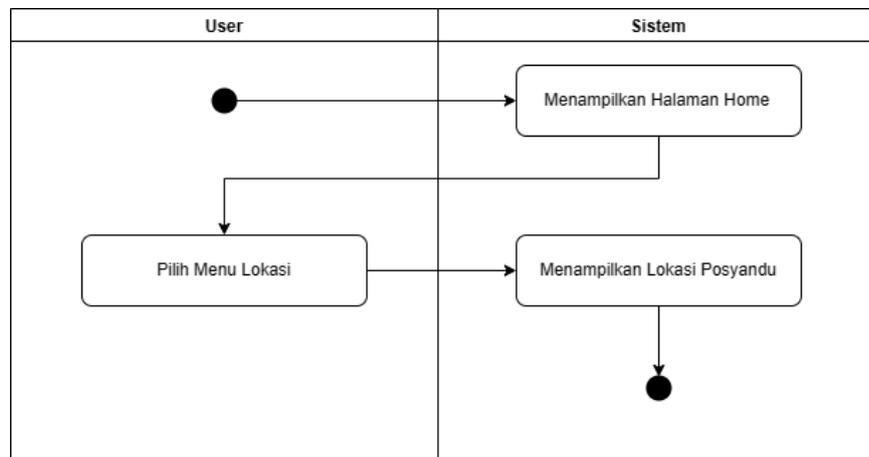
8) Activity Diagram menampilkan pembelajaran batita



Gambar 4. 12 *Activity* Diagram Pembelajaran Batita

Pada gambar 4.12 Menjelaskan cara pengguna dapat mengakses materi pembelajaran mengenai anak usia dini. Langkah pertama adalah pengguna membuka situs web yang kemudian akan menampilkan halaman beranda. Selanjutnya, pengguna dapat memilih opsi "Pembelajaran" yang terdapat di menu navigasi, dan sistem akan menampilkan pilihan antara materi untuk balita atau anak batita. Setelah itu, pengguna diminta untuk memilih salah satu dari materi pembelajaran yang tersedia, dan sistem akan menampilkan video interaktif sesuai dengan pilihan yang dibuat oleh pengguna.

9) Activity Diagram menampilkan lokasi posyandu



Gambar 4. 13 *Activity Diagram Menampilkan Lokasi Posyandu*

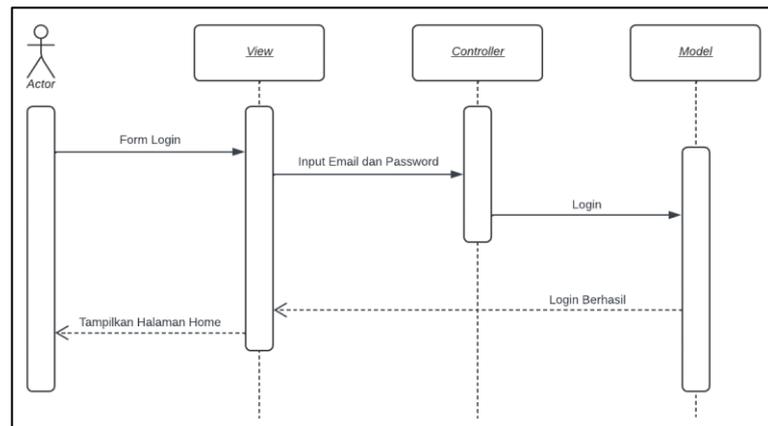
Pada gambar 4.13 Menjelaskan cara pengguna dapat mengakses lokasi posyandu yang tersedia. Langkah pertama adalah pengguna membuka situs web yang kemudian akan menampilkan halaman beranda. Selanjutnya, pengguna dapat memilih opsi "lokasi" yang terdapat di menu navigasi, dan sistem akan menampilkan maps yang menampilkan lokasi posyandu.

2. Sequence Diagram

Sequence Diagram adalah salah satu jenis diagram Interaksi yang menggambarkan cara suatu operasi dilaksanakan, termasuk pesan atau *request* yang dikirim dan waktu pelaksanaannya atau *respons*.

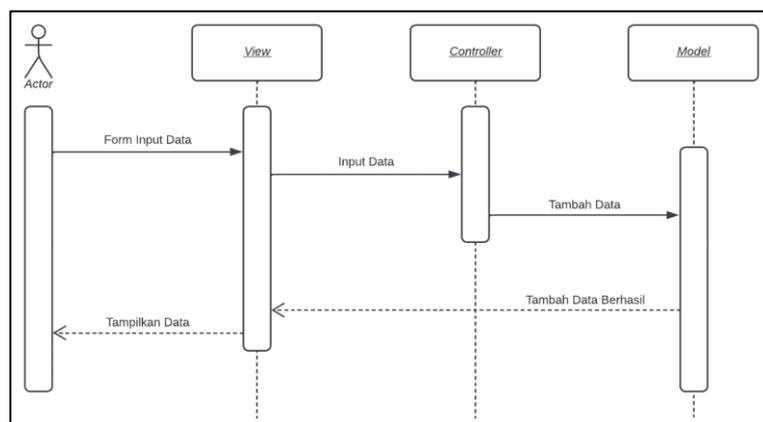
a. Sequence Diagram *Admin*

1) Sequence diagram Login



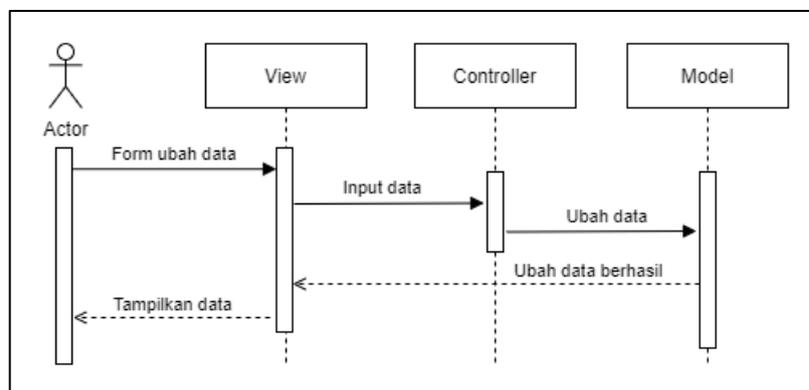
Gambar 4. 14 *Sequence Diagram Login*

2) Sequence diagram Tambah Data



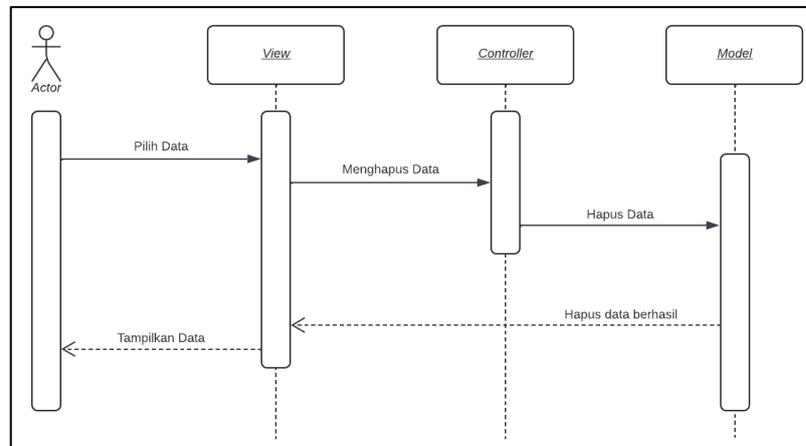
Gambar 4. 15 *Sequence Diagram Tambah Data*

3) Sequence diagram Ubah Data

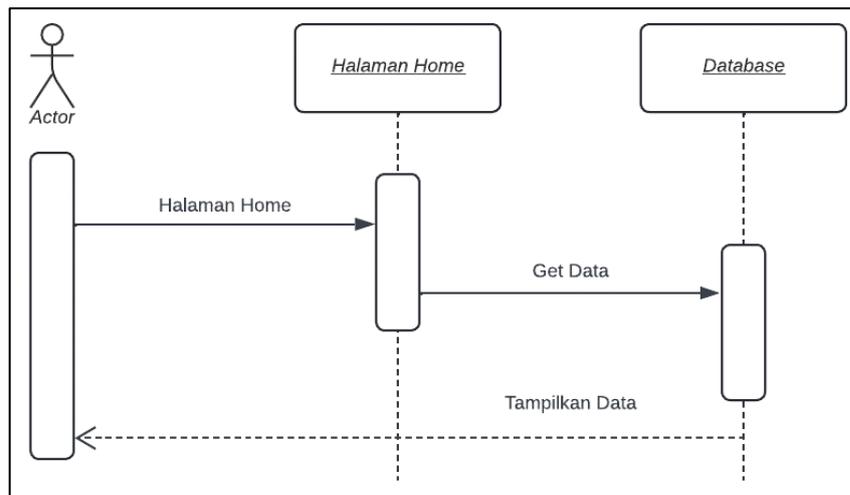


Gambar 4. 16 *Sequence Diagram Ubah Data*

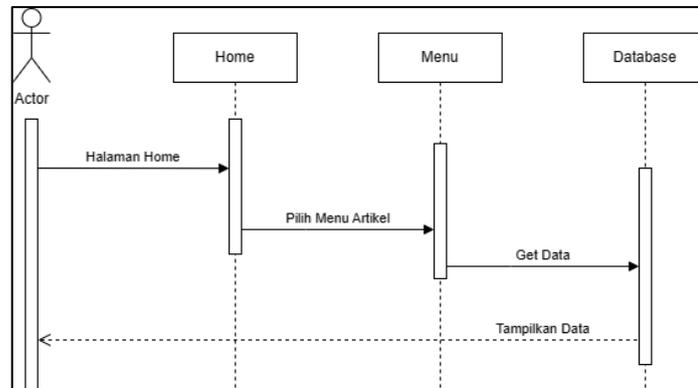
4) Sequence diagram hapus Data

Gambar 4. 17 *Sequence Diagram Hapus Data*b. Sequence Diagram *User*

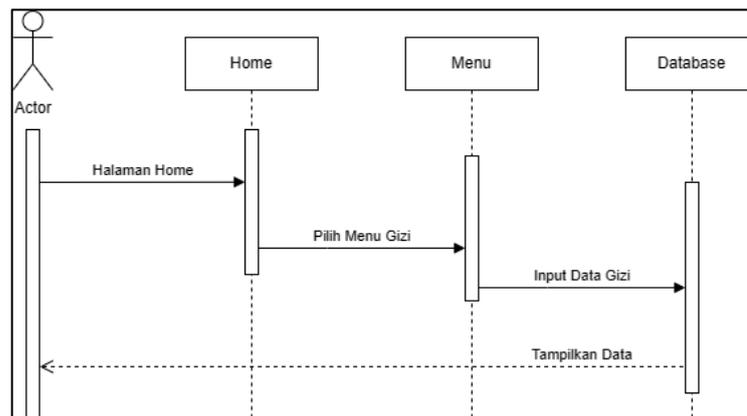
1) Sequence Diagram Home

Gambar 4. 18 *Sequence Diagram Artikel*

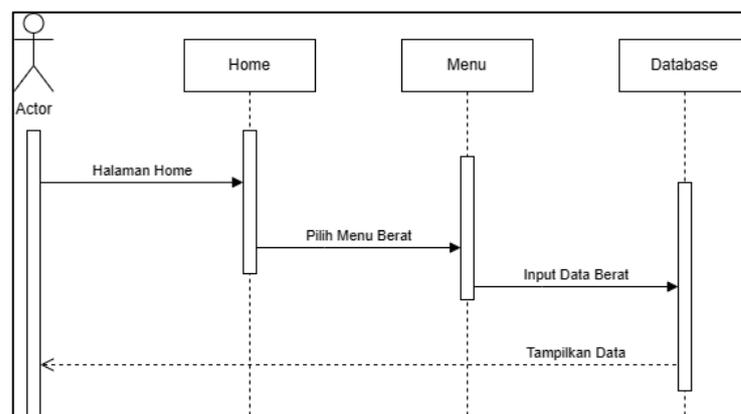
2) Sequence Diagram Artikel

Gambar 4. 19 *Sequence Diagram Artikel*

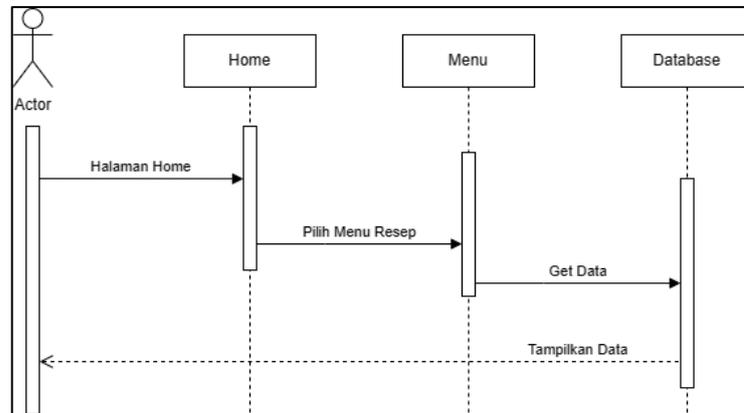
3) Sequence Diagram Cek Gizi

Gambar 4. 20 *Sequence Diagram Cek Gizi*

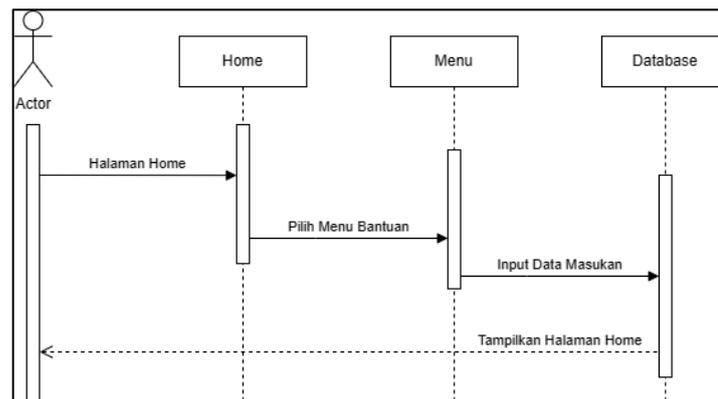
4) Sequence Diagram Cek Berat Badan

Gambar 4. 21 *Sequence Diagram Cek Berat Badan*

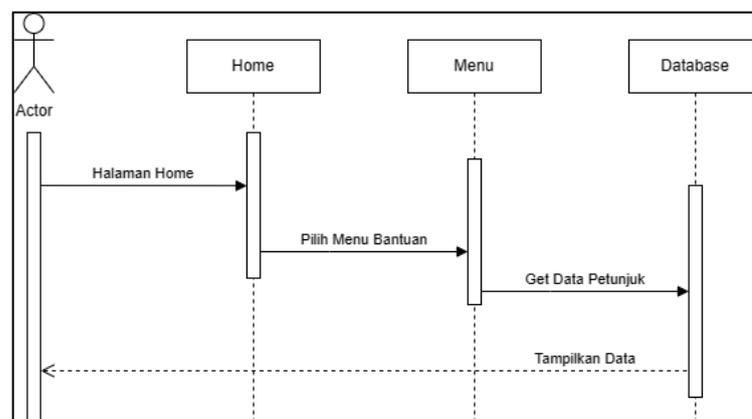
5) Sequence Diagram Resep

Gambar 4. 22 *Sequence Diagram Resep*

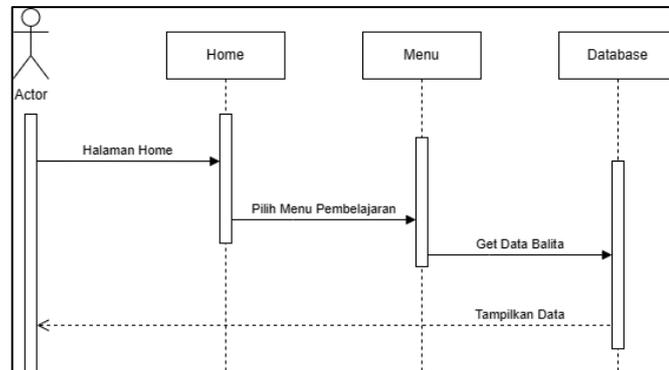
6) Activity Diagram Kirim Masukan

Gambar 4. 23 *Sequence Diagram Kirim Masukan*

7) Sequence Diagram Petunjuk Penggunaan

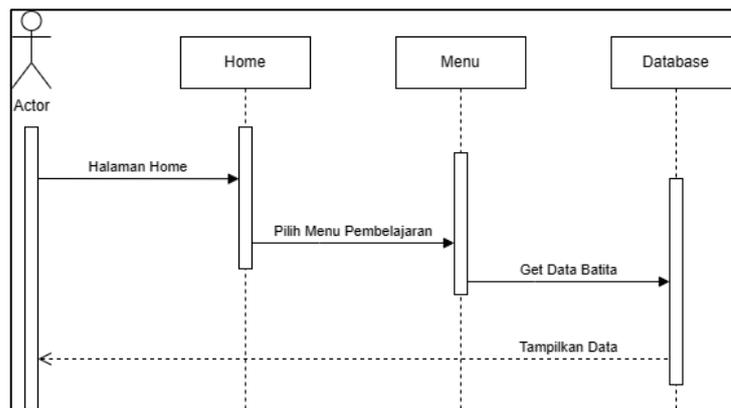
Gambar 4. 24 *Sequence Diagram Petunjuk Penggunaan*

8) Sequence Diagram Pembelajaran Balita



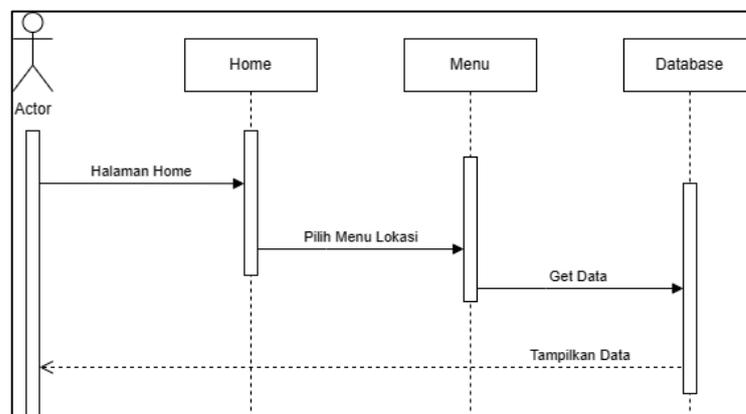
Gambar 4. 25 Sequence Diagram Pembelajaran Balita

9) Sequence Diagram cari Pembelajaran Batita



Gambar 4. 26 Sequence Diagram Pembelajaran Batita

10) Sequence Diagram cari lokasi Posyandu



Gambar 4. 27 Sequence Diagram Halaman Lokasi

B. Rancangan Database

Berikut adalah rancangan database untuk aplikasi media interaktif untuk mengetahui tumbuh kembang anak berdasarkan pengelolaan posyandu

Tabel 4. 1 *database user*

No	Nama	Jenis
1	<i>Id</i>	<i>Bigint unsigned</i>
2	<i>Name</i>	<i>Varchar(255)</i>
3	<i>Email</i>	<i>Varchar(255)</i>
4	<i>Password</i>	<i>Varchar(255)</i>
5	<i>Cerated_t</i>	<i>timestamp</i>

Tabel 4. 2 *Database artikels*

No	Nama	Jenis
1	<i>Id</i>	<i>Bigint unsigned</i>
2	<i>Judul</i>	<i>Vachar(255)</i>
3	<i>Penulis</i>	<i>Varchar(255)</i>
4	<i>Tanggal_posting</i>	<i>Date</i>
5	<i>Konten</i>	<i>Text</i>
6	<i>User_id</i>	<i>Bigint unsigned</i>
7	<i>Foto</i>	<i>Varchar(255)</i>
8	<i>Public</i>	<i>Tinyint(1)</i>
9	<i>Created_at</i>	<i>Timestamp</i>
10	<i>Update_at</i>	<i>timestamp</i>

Tabel 4. 3 *Database artikels*

No	Nama	Jenis
1	<i>id</i>	<i>Bigint unsigned</i>
2	<i>NamaObat</i>	<i>Varchar(255)</i>
3	<i>Pemberi</i>	<i>Varchar(255)</i>
4	<i>Tanggalpemberian</i>	<i>Date</i>
5	<i>Dosis</i>	<i>Text</i>
6	<i>Keterangan</i>	<i>Text</i>

7	<i>User_id</i>	<i>Bigint unsigned</i>
8	<i>Created_at</i>	<i>Timestamp</i>
9	<i>Update_at</i>	<i>timestamp</i>

Tabel 4. 4 Database feedback

No	Nama	Jenis
1	<i>id</i>	<i>Bigint unsigned</i>
2	<i>Pesan</i>	<i>Text</i>
3	<i>Dibaca</i>	<i>Text</i>
4	<i>Dibaca</i>	<i>Tinyint(1)</i>
5	<i>User_id</i>	<i>Bigint unsigned</i>
6	<i>Created_at</i>	<i>Timestamp</i>
7	<i>Update_at</i>	<i>timestamp</i>

Tabel 4. 5 Database model has roles

No	Nama	Jenis
1	<i>Role_id</i>	<i>Bigint unsigned</i>
2	<i>Model_type</i>	<i>Varhcar(255)</i>
3	<i>Model_id</i>	<i>Bigint unsigned</i>

Tabel 4. 6 Database roles

No	Nama	Jenis
1	<i>id</i>	<i>Bigint unsigned</i>
2	<i>Name</i>	<i>Varchar(255)</i>
3	<i>Guard_name</i>	<i>Varchar(255)</i>
4	<i>Created_at</i>	<i>Timestamp</i>
5	<i>Updated_at</i>	<i>timestamp</i>

Tabel 4. 7 Database Permissions

No	Nama	Jenis
1	<i>id</i>	<i>Bigint unsigned</i>
2	<i>Name</i>	<i>Varchar(255)</i>
3	<i>Guard_name</i>	<i>Varchar(255)</i>

4	<i>Created_at</i>	<i>Timestamp</i>
5	<i>Update_at</i>	<i>Timestamp</i>

Tabel 4. 8 Database failed jobs

No	Nama	Jenis
1	<i>id</i>	<i>Bigint unsigned</i>
2	<i>Uuid</i>	<i>Varchar(255)</i>
3	<i>Connection</i>	<i>Text</i>
4	<i>Queue</i>	<i>Text</i>
5	<i>Payload</i>	<i>Longtext</i>
6	<i>Exception</i>	<i>Logtext</i>
7	<i>Failed_at</i>	<i>Timestamp</i>

Tabel 4. 9 Database hasil status gizi

No	Nama	Jenis
1	<i>id</i>	<i>Bigint unsigned</i>
2	<i>User_id</i>	<i>Bigint unsigned</i>
3	<i>Tanggal</i>	<i>Date</i>
4	<i>Usia_bulan</i>	<i>Int</i>
5	<i>Berat_badan</i>	<i>Double(8.2)</i>
6	<i>Tinggi</i>	<i>Double(8.2)</i>
7	<i>Status_gizi</i>	<i>Varchar(255)</i>
8	<i>Created_at</i>	<i>Timestamp</i>
9	<i>Update_at</i>	<i>Timestamp</i>

Tabel 4. 10 Database migrations

No	Nama	Jenis
1	<i>id</i>	<i>Int unsigned</i>
2	<i>Migration</i>	<i>Varchar(255)</i>
3	<i>Batch</i>	<i>int</i>

Tabel 4. 11 Database Hasil berat badan

No	Nama	Jenis
1	<i>id</i>	<i>unsigned</i>
2	<i>User_id</i>	<i>Bigint unsigned</i>
3	<i>Berat_badan</i>	<i>Decimal(5.2)</i>
4	<i>Tanggal</i>	<i>Date</i>
5	<i>Hasil</i>	<i>Varchar(255)</i>
6	<i>Created_at</i>	<i>Timestamp</i>
7	<i>Update_at</i>	<i>timestamp</i>

Tabel 4. 12 Database password reset tokens

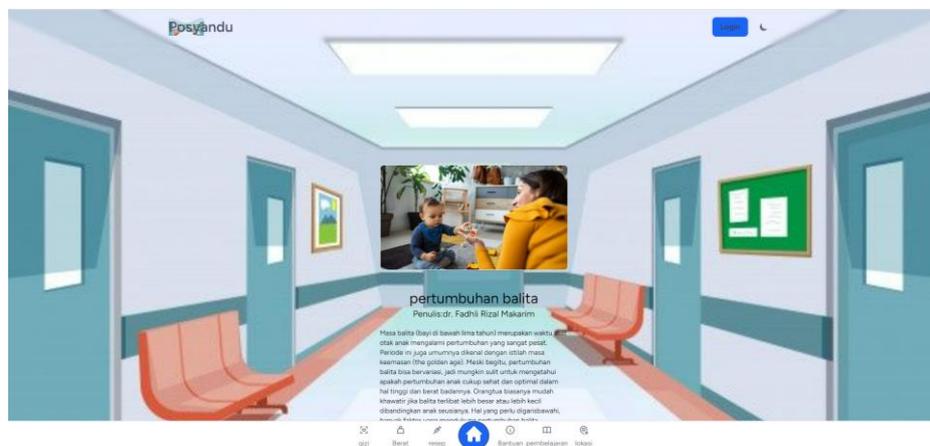
No	Nama	Jenis
1	<i>Email</i>	<i>Varchar(255)</i>
2	<i>Token</i>	<i>Varchar(255)</i>
3	<i>Created_at</i>	<i>timestamp</i>

C. Detail Sistem

1. Admin

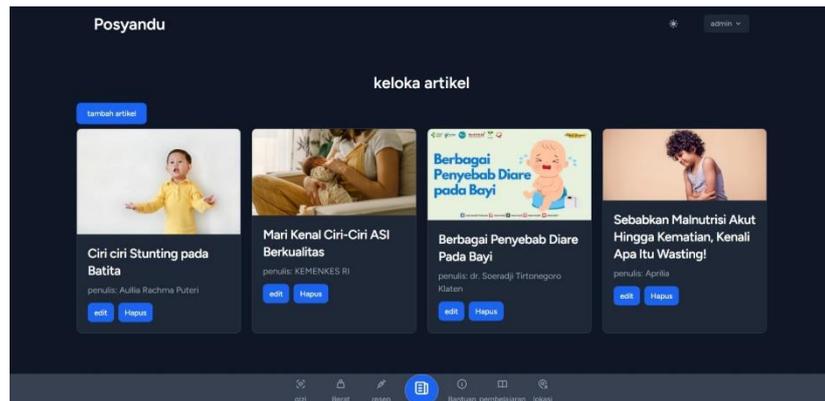
a. Halaman Login

Merupakan halaman awal yang digunakan *Admin* untuk mendapat akses masuk ke halaman menu *Admin*.

Gambar 4. 28 Halaman *Login Admin*

b. Halaman Dashboard *Admin*

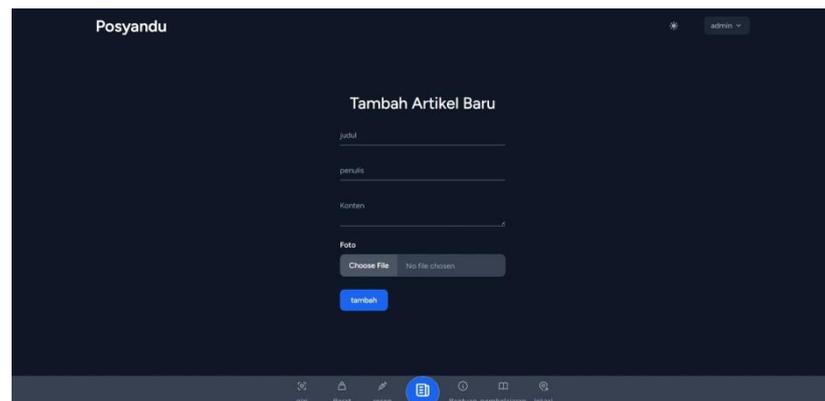
Merupakan halaman awal *Admin* yang berisi artikel-artikel yang ada pada website posyandu, dan dapat mengelola artikel baik untuk tambah artikel, edit artikel maupun hapus artikel.



Gambar 4. 29 Halaman *Dashboard Admin*

c. Halaman Tambah data artikel

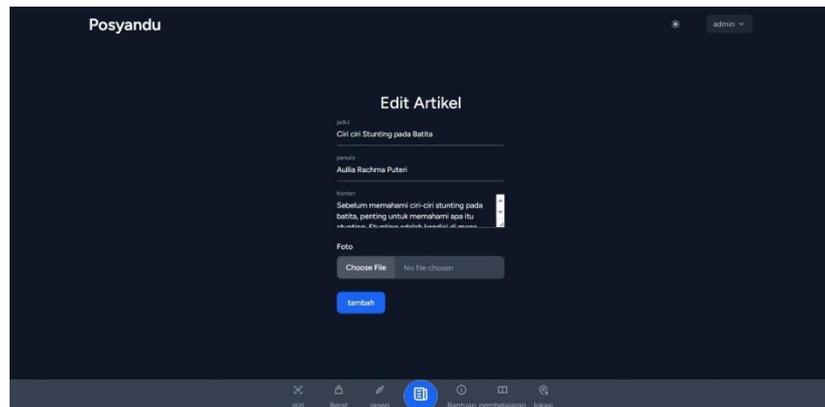
Merupakan halaman yang berisi form, berfungsi untuk menambah data artikel pada website



Gambar 4. 30 Halaman Tambah Data Artikel

d. Halaman Edit Data Artikel

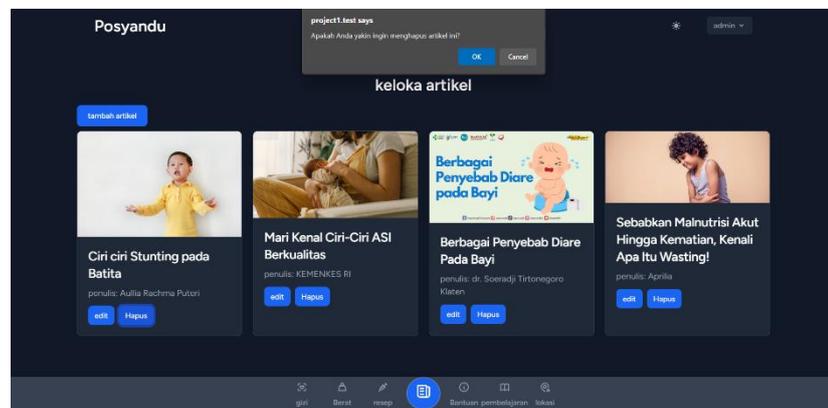
Merupakan halaman yang berisi form untuk melakukan pengeditan data artikel.



Gambar 4. 31 Halaman Edit Data Artikel

e. Halaman Hapus Data

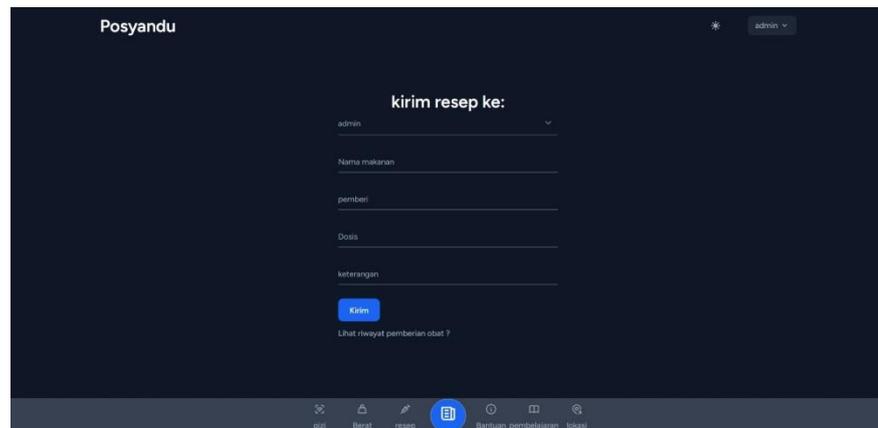
Merupakan halaman yang berisi peringatan untuk melakukan hapus data artikel.



Gambar 4. 32 Halaman Hapus Data Artikel

f. Halaman Resep

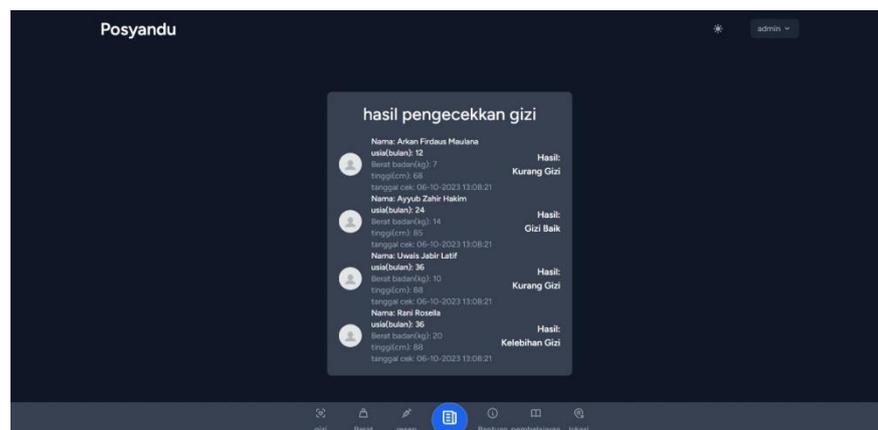
Merupakan halaman yang berisi form untuk melakukan penginputan resep untuk balita maupun batita dan dikirim ke *User*



Gambar 4. 33 Halaman Resep

g. Halaman Gizi

Merupakan halaman yang berisi informasi-informasi dari inputan dan hasil dari cek gizi *User*



Gambar 4. 34 Halaman Gizi

h. Halaman berat dan tinggi badan

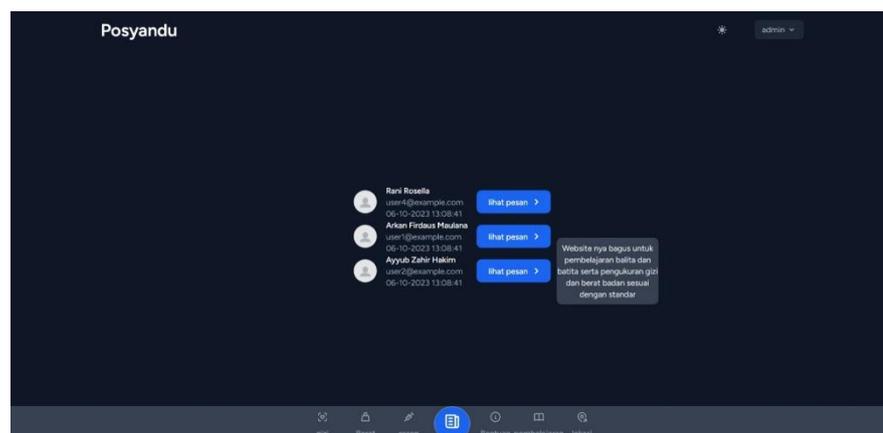
Merupakan halaman yang berisi informasi-informasi dari inputan dan hasil dari cek berat badan dan tinggi badan *User*



Gambar 4. 35 Halaman Berat dan Tinggi Badan

i. Halaman lihat feedback

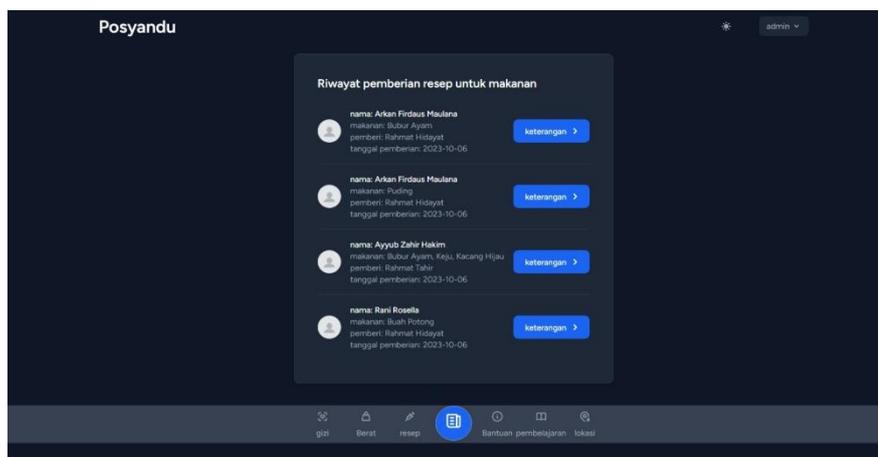
Merupakan halaman yang berisi informasi dari masukan atau *feedback* dari *User*



Gambar 4. 36 Halaman Lihat Feedback

j. Halaman riwayat resep/rekomendasi makanan

Merupakan halaman yang berisi informasi dari masukan atau kiriman resep atau rekomendasi dari petugas kesehatan maupun dokter untuk *User*.

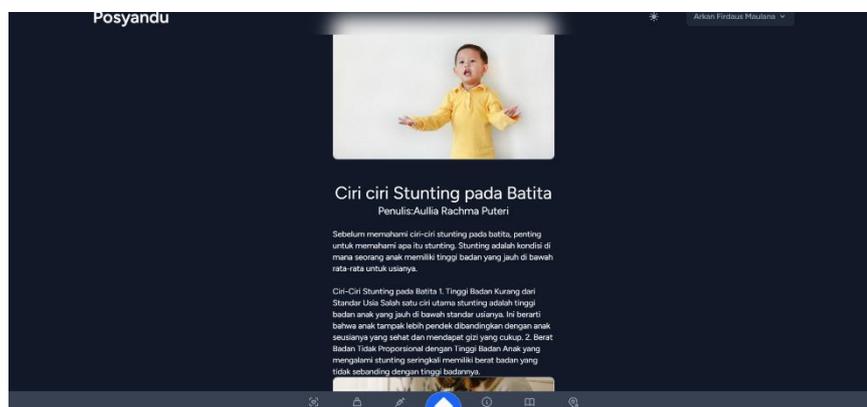


Gambar 4. 37 Halaman Riwayat Resep atau Rekomendasi Makanan

2. User

a. Halaman *Home Website*

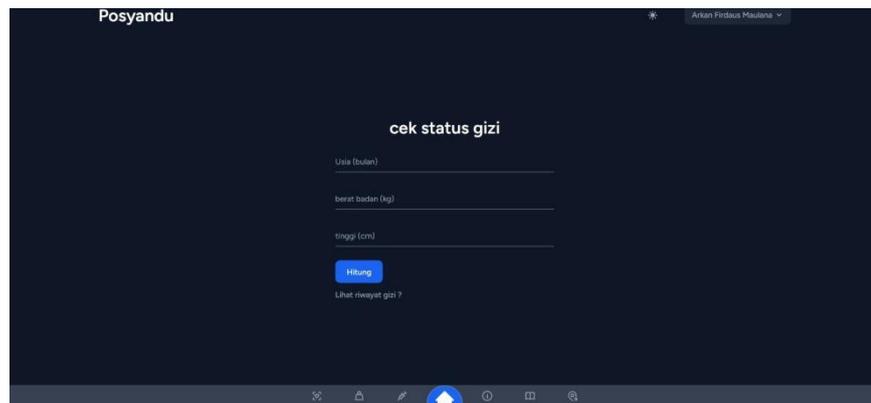
Merupakan halaman awal bagi *User* yang mengakses *website* Interaktif untuk Mengetahui Tumbuh Kembang Anak



Gambar 4. 38 Halaman Home Website

b. Halaman Cek Status Gizi

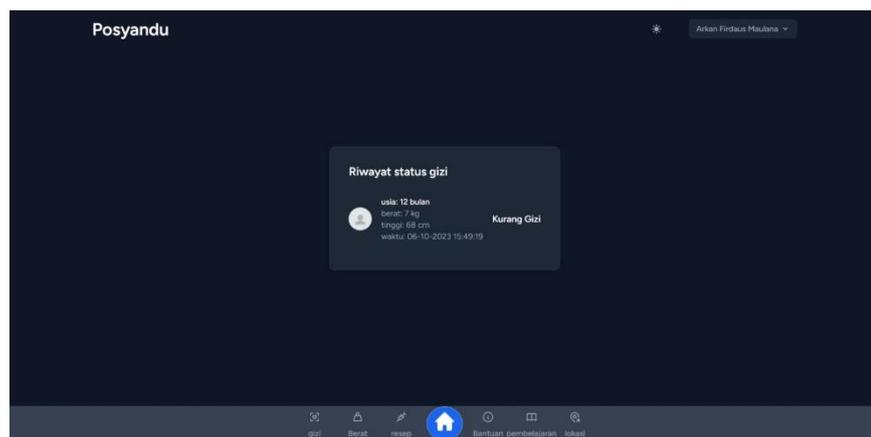
Merupakan halaman yang berfungsi untuk menampilkan form untuk mengecek status gizi.



Gambar 4. 39 Halaman Cek Status Gizi

c. Halaman Hasil Cek Status Gizi

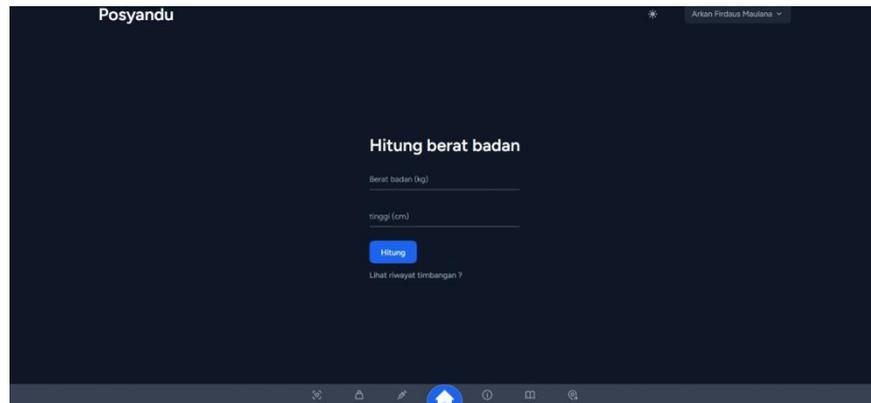
Merupakan halaman yang berfungsi untuk menampilkan data Hasil Cek Status Gizi.



Gambar 4. 40 Halaman Hasil Cek Status Gizi

d. Halaman Hitung Berat Badan

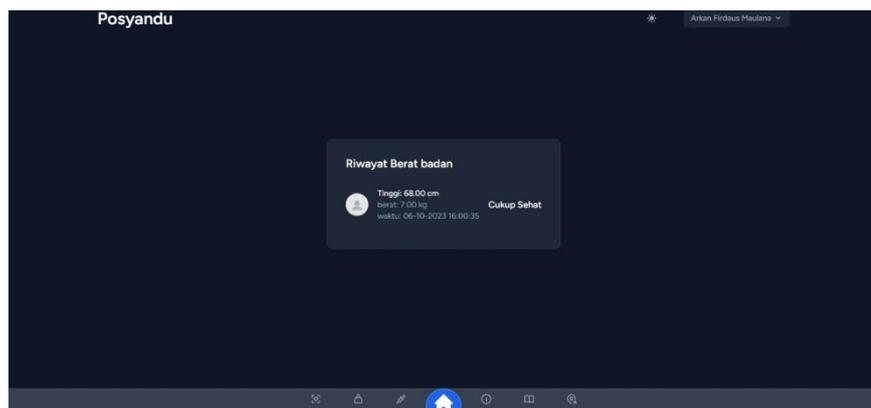
Merupakan halaman yang berfungsi untuk menampilkan form untuk menghitung berat badan.



Gambar 4. 41 Halaman Hitung Berat Badan

e. Halaman Hasil Hitung Berat Badan

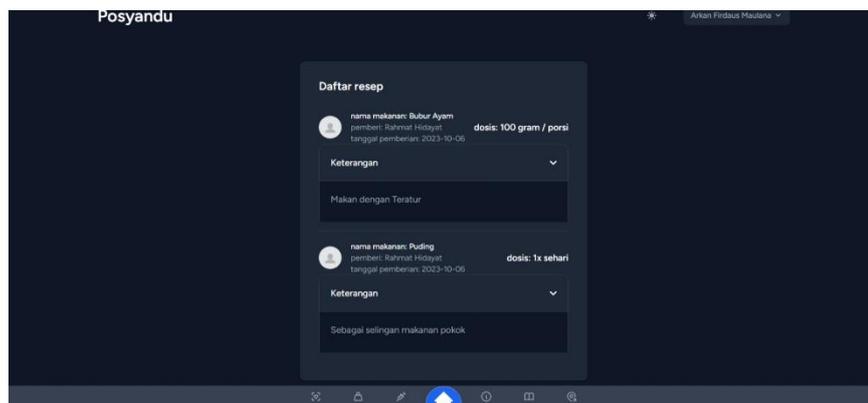
Merupakan halaman yang berfungsi untuk menampilkan data Hasil Hitung Berat Badan.



Gambar 4. 42 Halaman Hasil Hitung Berat Badan

f. Halaman Daftar Resep/Rekomendasi Makanan

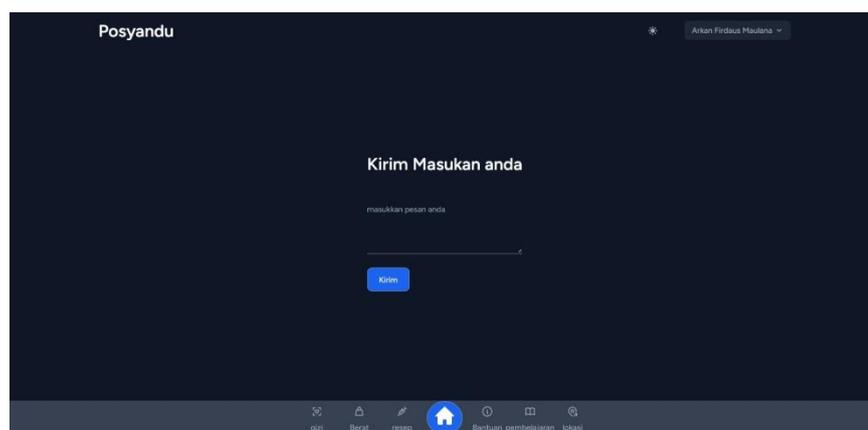
Merupakan halaman yang berfungsi untuk menampilkan data Daftar Resep/Rekomendasi Makanan.



Gambar 4. 43 Halaman Daftar Resep/Rekomendasi Makanan

g. Halaman Kirim Masukan

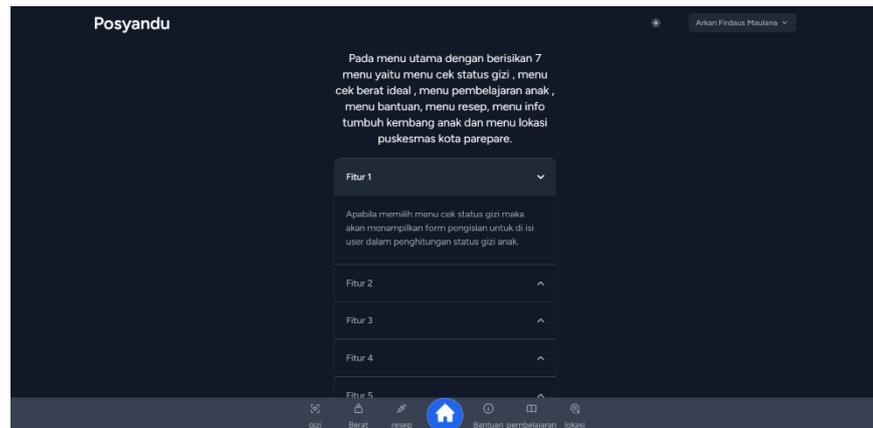
Merupakan halaman yang berfungsi untuk menampilkan form untuk mengirim masukan kepada dokter ataupun tenaga kesehatan.



Gambar 4. 44 Halaman Form Kirim Masukan

h. Halaman Petunjuk Penggunaan

Merupakan halaman yang berfungsi untuk menampilkan data petunjuk penggunaan *website*.



Gambar 4. 45 Halaman Petunjuk Penggunaan

i. Halaman Pembelajaran pengetahuan tentang Balita

Merupakan halaman yang berfungsi untuk menampilkan data berupa video interaktif pengetahuan tentang balita.



Gambar 4. 46 Halaman Pembelajaran Tentang Balita

j. Halaman Pembelajaran Pengetahuan Tentang Batita

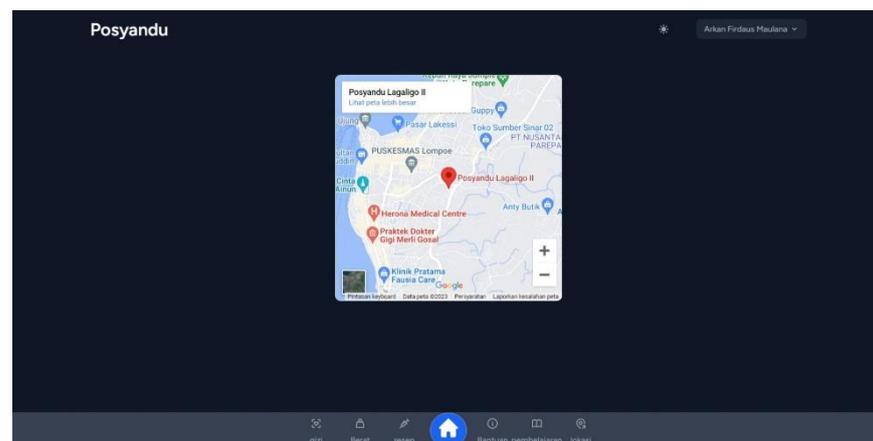
Merupakan halaman yang berfungsi untuk menampilkan data berupa video interaktif pengetahuan tentang balita.



Gambar 4. 47 Halaman Pembelajaran Tentang Batita

k. Halaman Lokasi Posyandu

Merupakan halaman yang berfungsi untuk menampilkan data lokasi Posyandu.



Gambar 4. 48 Halaman Lokasi Posyandu

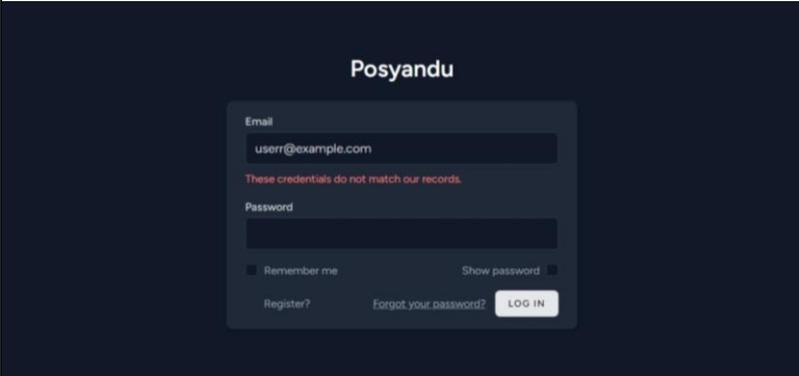
D. Pengujian Sistem

Pada penelitian ini, sistem diuji menggunakan dua metode, yaitu *Black Box Testing* dan *White Box Testing*. Berikut adalah hasil pengujian dengan menggunakan kedua metode tersebut.

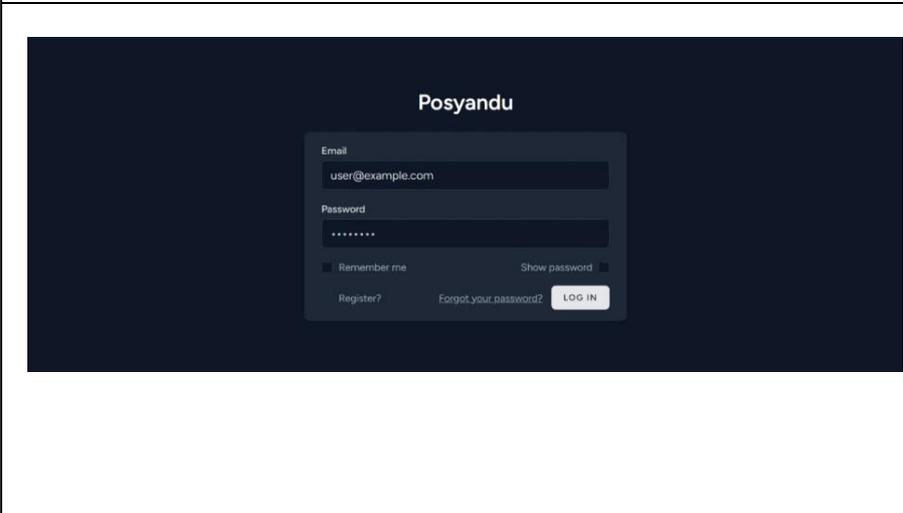
1. *Black Box*

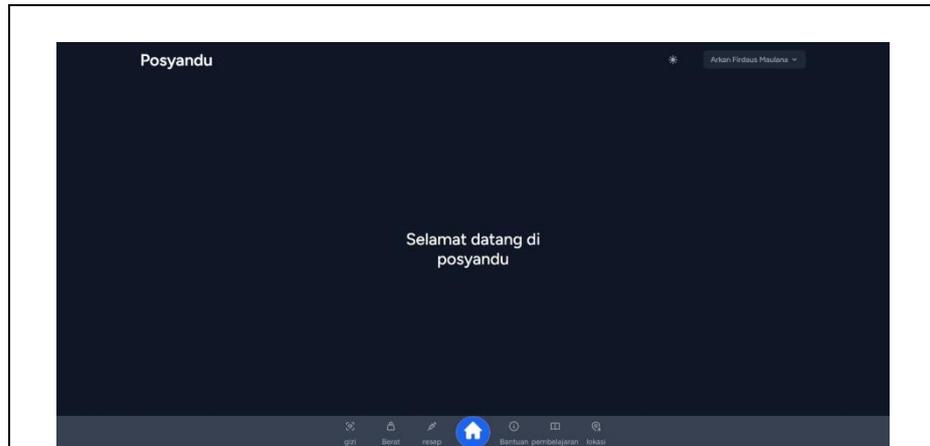
a. *Black Box Testing* Kesalahan *Email* dan *Password*

Tabel 4. 13 *Black Box* Testing Kesalahan *Email* dan *Password*

Tes Faktor	Hasil	Keterangan
Memasukkan <i>Email</i> atau <i>Password</i> yang tidak sesuai	✓	Berhasil, ketika <i>Email</i> atau <i>Password</i> tidak sesuai maka tampil pesan <i>These credentials do not match our records.</i>
Screenshot		
		

b. *Black Box* Testing Login BerhasilTabel 4. 14 *Black Box* Testing Login Berhasil

Tes Faktor	Hasil	Keterangan
Memasukkan <i>Email</i> atau <i>Password</i> yang benar	✓	Berhasil, sistem dapat menampilkan halaman <i>Admin/dashbord Admin</i>
Screenshot		
		

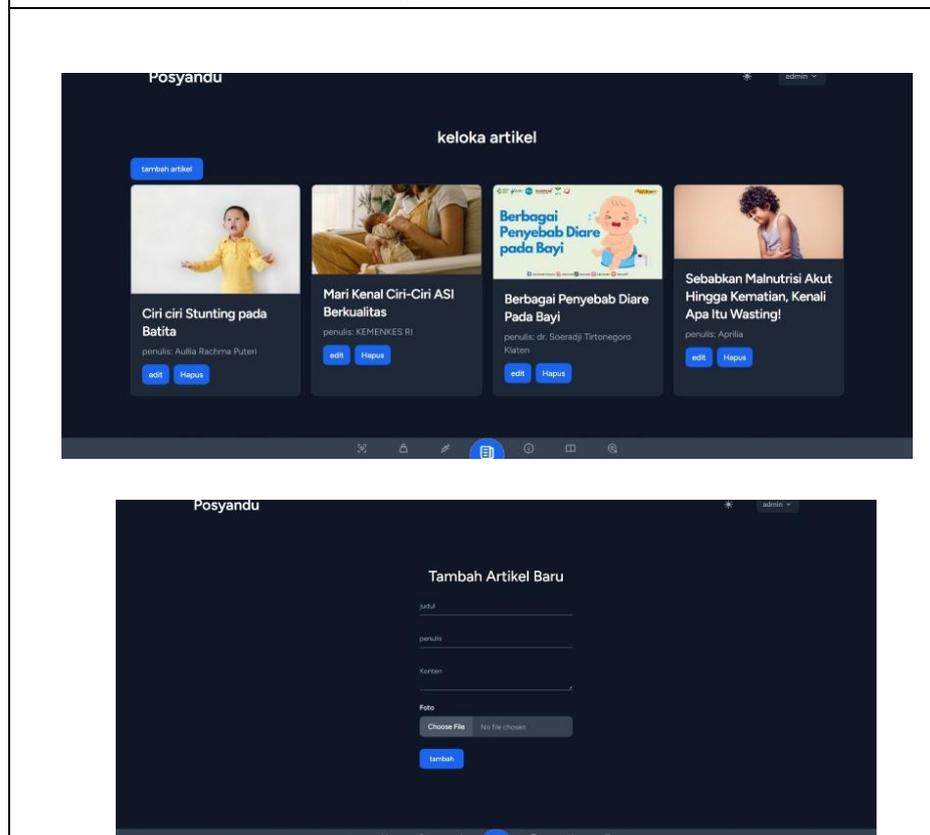


c. *Black Box Testing* Tambah Data

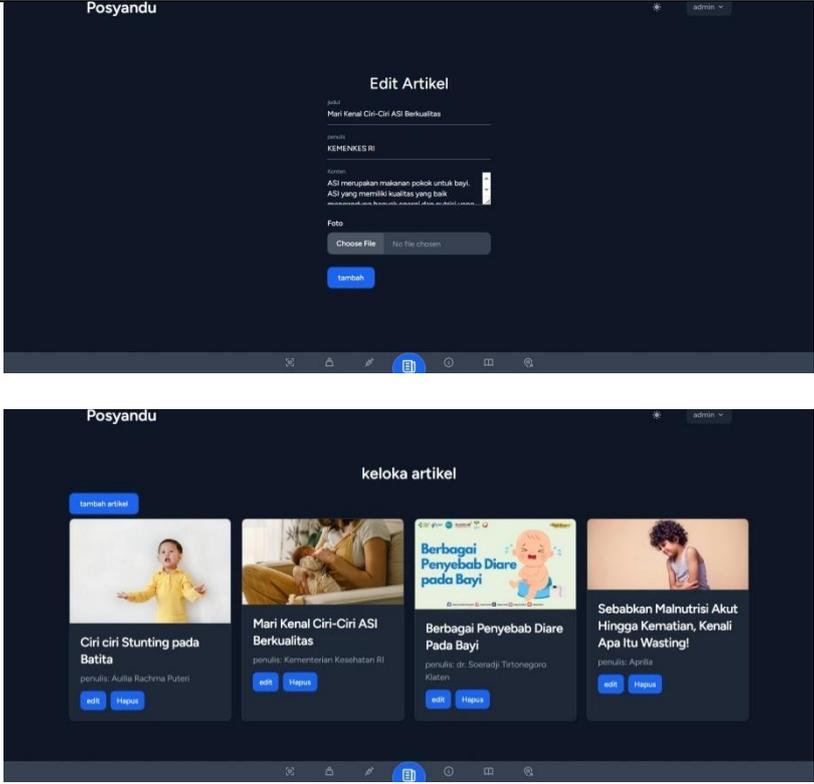
Tabel 4. 15 *Black Box* Testing Tambah Data

Tes Faktor	Hasil	Keterangan
Admin mengisi form tambah dan menekan tombol simpan	✓	Berhasil, tampil data yang telah berhasil ditambahkan

Screenshot

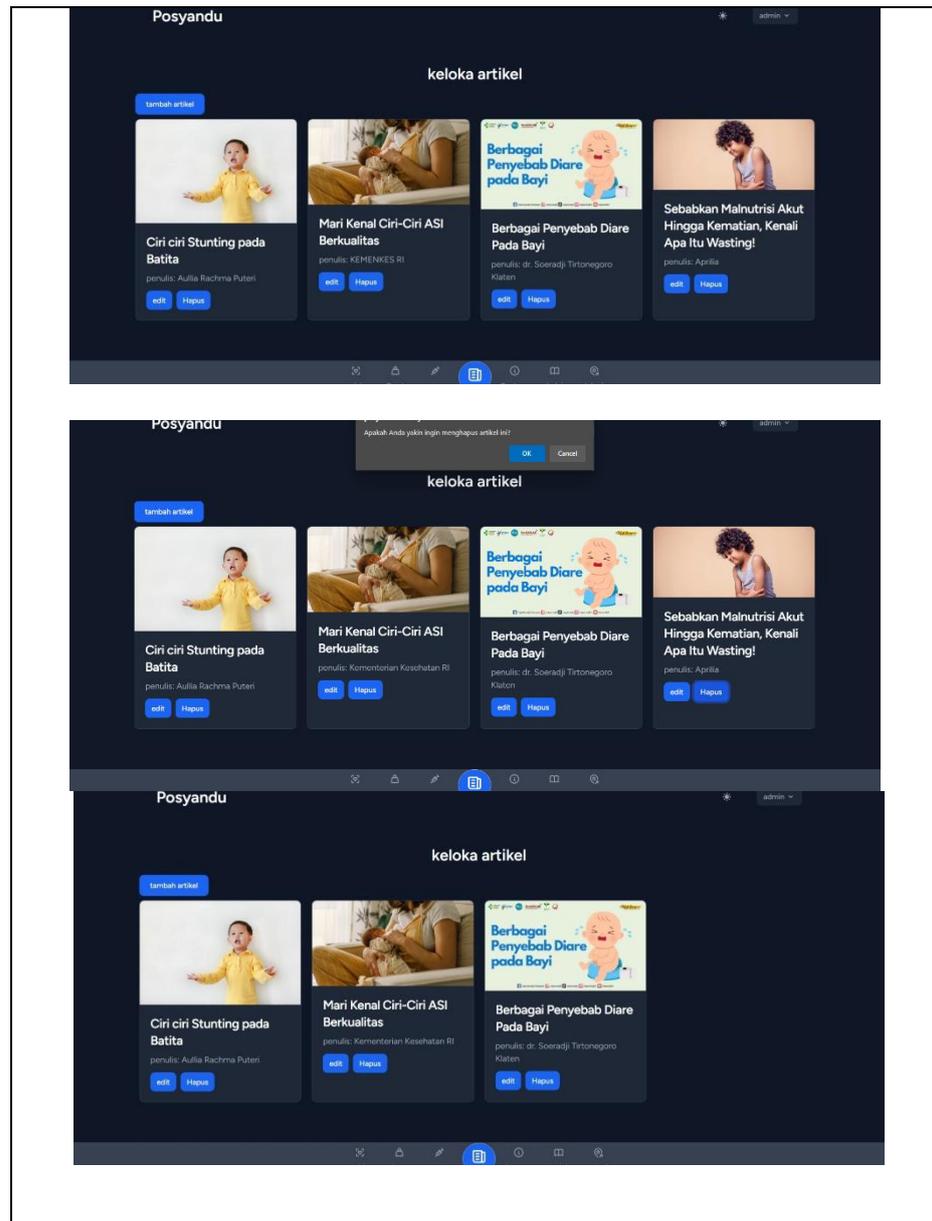


d. *Black Box Testing* Ubah DataTabel 4. 16 *Black Box* Testing Ubah Data

Tes Faktor	Hasil	Keterangan
<i>Admin</i> mengisi form ubah data dan menekan tombol <i>update</i>	✓	Berhasil, tampil data telah berhasil diubah
Screenshot		
		

e. *Black Box Testing* Hapus DataTabel 4. 17 *Black Box* Testing Hapus Data

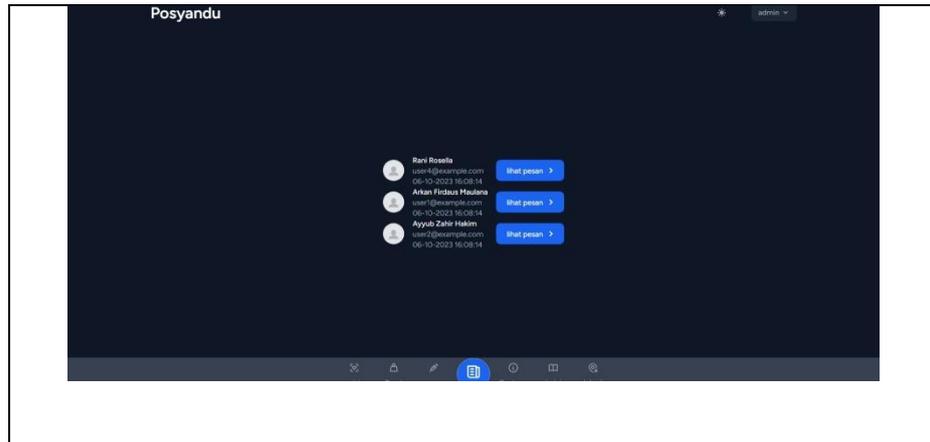
Tes Faktor	Hasil	Keterangan
<i>Admin</i> menghapus salah satu data	✓	Berhasil, data artikel telah dihapus
Screenshot		



f. *Black Box Testing* Hapus Data

Tabel 4. 18 *Black Box Testing* Hapus Data

Tes Faktor	Hasil	Keterangan
<i>Admin</i> melihat masukan yang telah dikirim <i>User</i>	✓	Berhasil, tampil data masukan dari <i>User</i>
<i>Screenshot</i>		



g. *Black Box Testing Halaman Home Website*

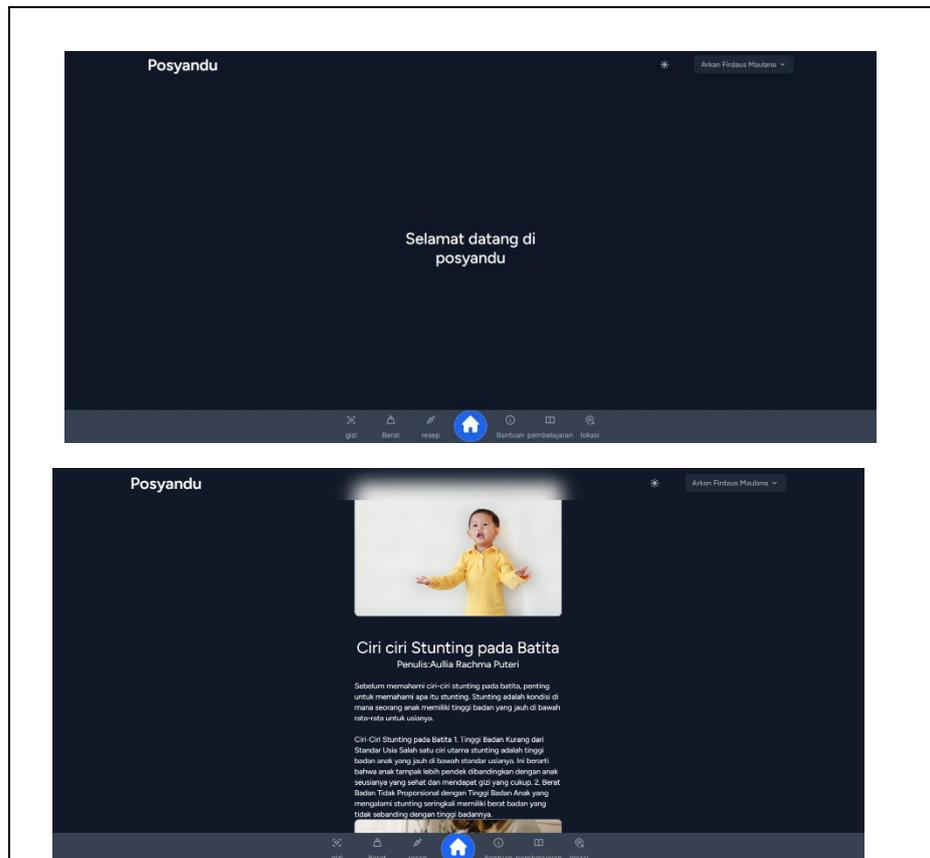
Tabel 4. 19 *Black Box Testing Halaman Home Website*

Tes Faktor	Hasil	Keterangan
<i>User pertama kali mengakses website</i>	✓	Berhasil, Tampil Halaman Home
Screenshot		

h. *Black Box Testing Halaman Artikel*

Tabel 4. 20 *Black Box Testing Halaman Artikel*

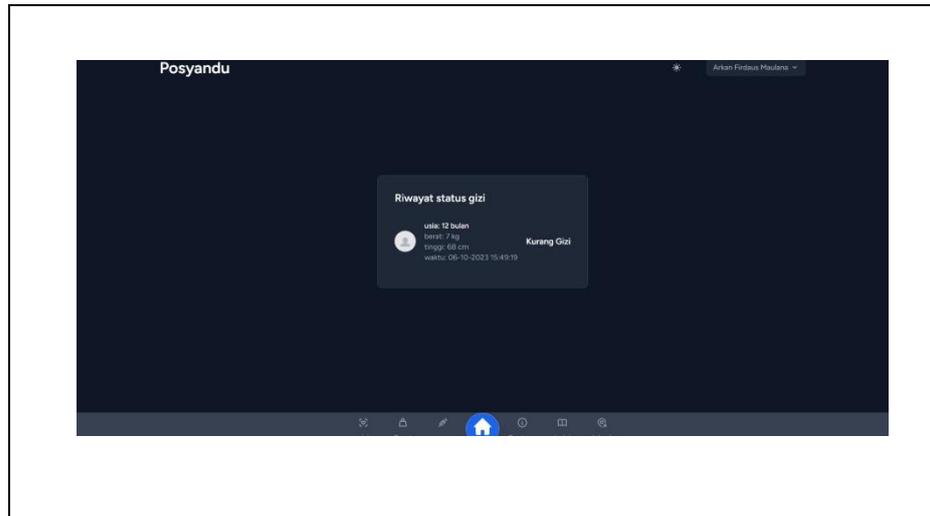
Tes Faktor	Hasil	Keterangan
<i>User menekan tombol home dibagian navbar</i>	✓	Berhasil, Tampil beberapa artikel yang telah ditambahkan
Screenshot		



i. *Black Box Testing* Halaman Cek Status Gizi

Tabel 4. 21 *Black Box Testing* Halaman Cek Status Gizi

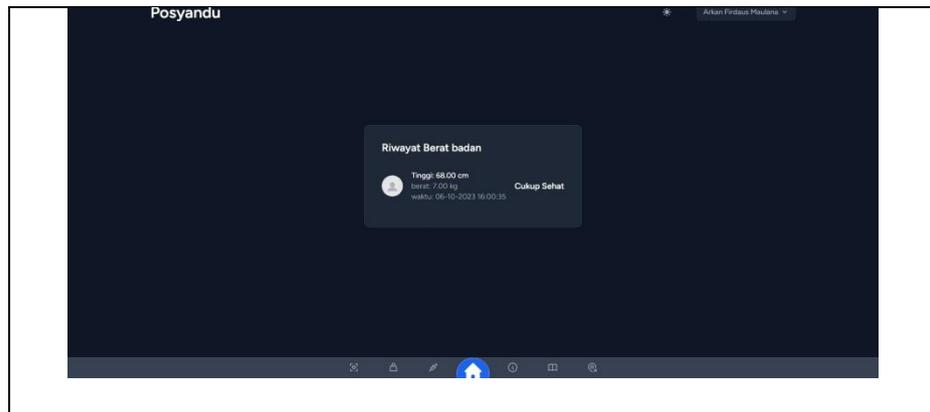
Tes Faktor	Hasil	Keterangan
User menekan tombol gizi dibagian navbar	✓	Berhasil, Tampil Halaman Cek Status Gizi berisi form isian cek gizi
Screenshot		



j. *Black Box Testing* Halaman Cek Berat dan Tinggi Badan

Tabel 4. 22 *Black Box Testing* Cek Berat dan Tinggi Badan

Tes Faktor	Hasil	Keterangan
<i>User</i> menekan tombol berat dibagian navbar	✓	Berhasil, Tampil Halaman Hitung Berat dan Tinggi Badan berisi form isian cek berat dan tinggi badan.
Screenshot		



k. *Black Box Testing* Halaman Semua *Datasheet*

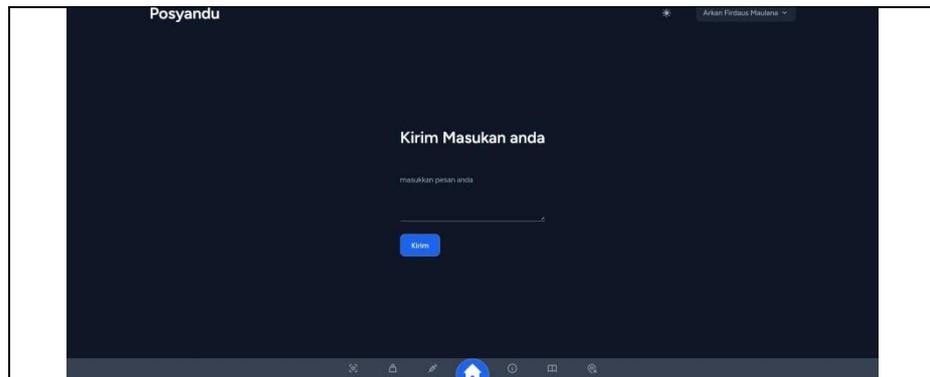
Tabel 4. 23 *Black Box Testing* Halaman Semua *Datasheet*

Tes Faktor	Hasil	Keterangan
<i>User</i> menekan tombol resep dibagian navbar	✓	Berhasil, Tampil Semua Data Resep/Rekomendasi Makanan dari Dokter maupun Tenaga Kesehatan.
Screenshot		

l. *Black Box Testing* Halaman Kirim masukan

Tabel 4. 24 *Black Box Testing* Halaman Kirim Masukan

Tes Faktor	Hasil	Keterangan
<i>User</i> menekan tombol bantuan kemudian memilih menu kirim masukan dibagian navbar	✓	Berhasil, Tampil <i>form</i> kirim masukan
Screenshot		



m. *Black Box Testing* Petunjuk Penggunaan

Tabel 4. 25 *Black Box Testing* Petunjuk Penggunaan

Tes Faktor	Hasil	Keterangan
<i>User</i> menekan tombol bantuan kemudian memilih petunjuk penggunaan dibagian navbar	✓	Berhasil, Tampil data petunjuk penggunaan fitur dalam <i>website</i>
Screenshot		

n. *Black Box Testing* Pembelajaran Balita

Tabel 4. 26 *Black Box Testing* Pembelajaran Balita

Tes Faktor	Hasil	Keterangan
<i>User</i> menekan tombol pembelajaran kemudian memilih menu balita dibagian navbar	✓	Berhasil, Tampil multimedia interaktif berupa video
Screenshot		



o. *Black Box Testing* Pembelajaran Batita

Tabel 4. 27 *Black Box Testing* Pembelajaran Batita

Tes Faktor	Hasil	Keterangan
User menekan tombol pembelajaran kemudian memilih menu batita dibagian navbar	✓	Berhasil, Tampil multimedia interaktif berupa video

Screenshot

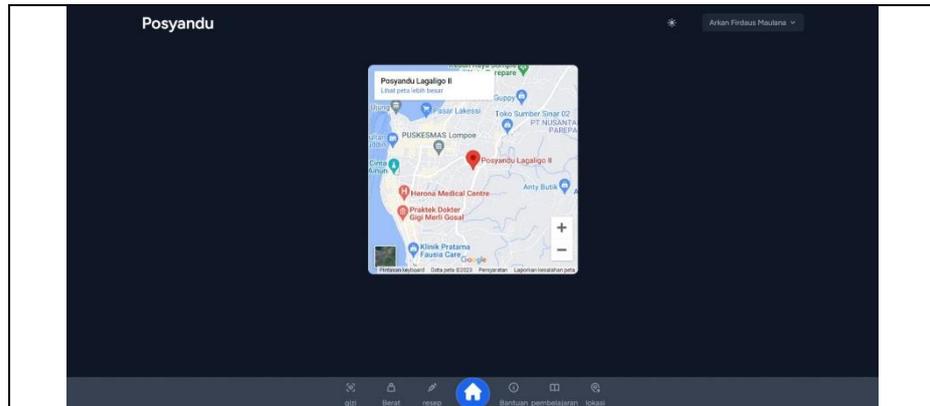


p. *Black Box Testing* Lokasi Posyandu

Tabel 4. 28 *Black Box Testing* Lokasi Posyandu

Tes Faktor	Hasil	Keterangan
User menekan tombol Lokasi dibagian navbar	✓	Berhasil, Map atau peta lokasi Posyandu

Screenshot

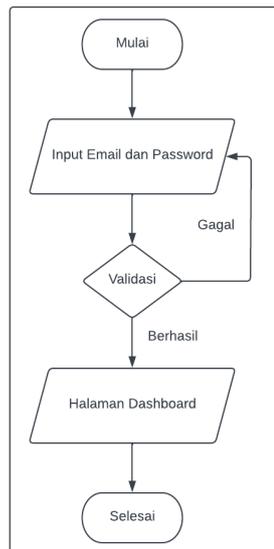


2. White Box

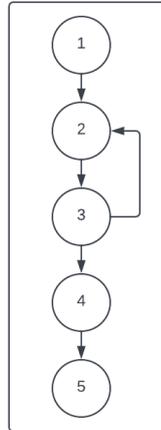
Pada metode pengujian ini akan ditampilkan *Flowchart* dan *Flowgraph* dari sistem yang telah dibuat. Berikut dibawah ini merupakan hasil pengujian menggunakan metode *White Box Testing*.

a. *White Box Testing* Kesalahan *Email* dan *Password*

1) *Flowchart*



Gambar 4. 49 *Flowchart* Kesalahan *Email* dan *Password*

2) *Flowgraph*Gambar 4. 50 *Flowgraph* Kesalahan *Email* dan *Password*

Berdasarkan gambar 4. 50 diatas dilakukan perhitungan sebagai berikut :

(1) Menghitung *Cyclomatic Complexity* $V(G)$ pada *egde* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 5$$

$$N \text{ (node)} = 5$$

$$P \text{ (Predikat node)} = 1$$

Penyelesaian :

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

$$= 5 - 5 + 2$$

$$= 2$$

$$\text{Predikat (P)} = P + 1$$

$$= 1 + 1$$

$$= 2$$

(2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* di atas memiliki *Region* = 2

(3) *Independent path* pada *Flowgraph* tersebut yakni:

Path 1 = 1 – 2 – 3 – 2

Path 2 = 1 – 2 – 3 – 4 – 5

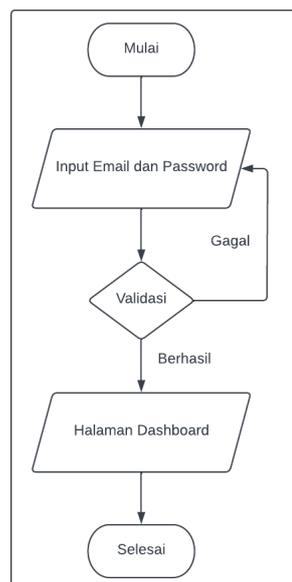
(4) *Grafik matriks* kesalahan *Email* dan *Password*

Tabel 4. 29 *Grafik matriks* Kesalahan *Email* dan *Password*

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

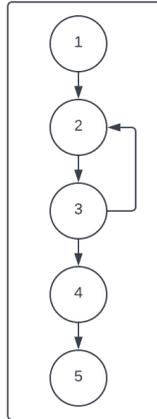
b. *White Box Testing* Login Berhasil

1) *Flowchart*



Gambar 4. 51 *Flowchart* Login Berhasil

2) *Flowgraph*



Gambar 4. 52 *Flowgraph* Login Berhasil

Berdasarkan gambar 4. 52 diatas dilakukan perhitungan sebagai berikut :

(1) Menghitung *Cyclomatic Complexity* $V(G)$ pada *egde* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 5$$

$$N \text{ (node)} = 5$$

$$P \text{ (Predikat node)} = 1$$

Penyelesaian :

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

$$= 5 - 5 + 2$$

$$= 2$$

$$\text{Predikat (P)} = P + 1$$

$$= 1 + 1$$

$$= 2$$

(2) Berdasarkan perhitungan *Cyclomatic Complexity* dari

Flowgraph di atas memiliki Region = 2

(3) *Independent path* pada *Flowgraph* tersebut yakni:

Path 1 = 1 – 2 – 3 – 2

Path 2 = 1 – 2 – 3 – 4 – 5

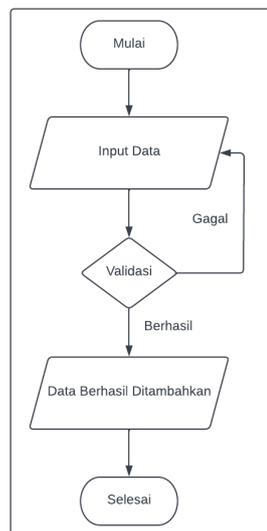
(4) *Grafik matriks* login berhasil

Tabel 4. 30 Grafik matriks *Login Berhasil*

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

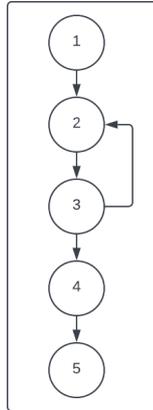
c. *White Box Testing* Tambah Data Artikel

1) *Flowchart*



Gambar 4. 53 *Flowchart* Tambah Data

2) *Flowgraph*



Gambar 4. 54 *Flowgraph* Tambah Data
Berdasarkan gambar 4. 54 diatas dilakukan perhitungan sebagai

berikut :

(1) Menghitung *Cyclomatic Complexity* $V(G)$ pada *egde* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 5$$

$$N \text{ (node)} = 5$$

$$P \text{ (Predikat node)} = 1$$

Penyelesaian :

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

$$= 5 - 5 + 2$$

$$= 2$$

$$\text{Predikat (P)} = P + 1$$

$$= 1 + 1$$

$$= 2$$

(2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* di atas memiliki *Region* = 2

(3) *Independent path* pada *Flowgraph* tersebut yakni:

Path 1 = 1 - 2 - 3 - 2

Path 2 = 1 - 2 - 3 - 4 - 5

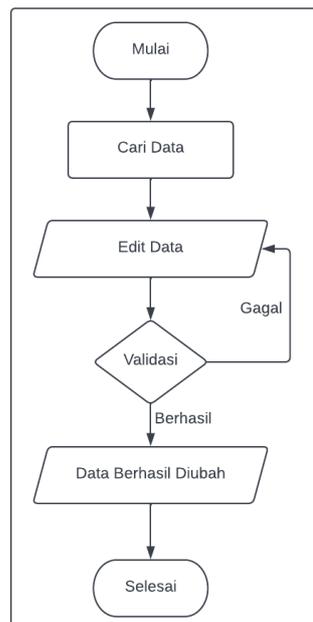
(4) *Grafik matriks* tambah data

Tabel 4. 31 Grafik matriks Tambah Data

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

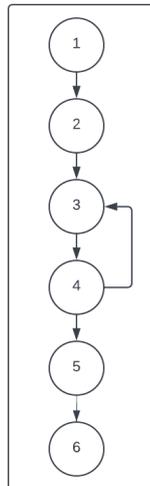
d. *White Box Testing* Ubah Data Artikel

1) *Flowchart*



Gambar 4. 55 *Flowchart* Ubah Data

2) *Flowgraph*



Gambar 4. 56 *Flowgraph* Ubah Data

Berdasarkan gambar 4. 56 diatas dilakukan perhitungan sebagai berikut :

(1) Menghitung *Cyclomatic Complexcity* $V(G)$ pada *egde* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 6$$

$$N \text{ (node)} = 6$$

$$P \text{ (Predikat node)} = 1$$

Penyelesaian :

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

$$= 6 - 6 + 2$$

$$= 2$$

$$\text{Predikat (P)} = P + 1$$

$$= 1 + 1$$

$$= 2$$

(2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* di atas memiliki *Region* = 2

(3) *Independent path* pada *Flowgraph* tersebut yakni:

Path 1 = 1 – 2 – 3 – 4 – 3

Path 2 = 1 – 2 – 3 – 4 – 5 – 6

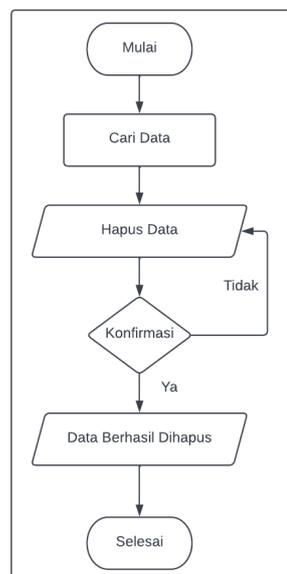
(4) *Grafik matriks* ubah data

Tabel 4. 32 Grafik matriks Ubah Data

	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	
	SUM (E + 1)							$1 + 1 = 2$

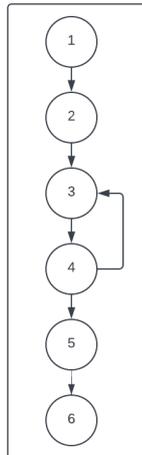
e. *White Box Testing* Hapus Data Artikel

1) *Flowchart*



Gambar 4. 57 *Flowchart* Hapus Data

2) *Flowgraph*



Gambar 4. 58 *Flowgraph* Hapus Data

Berdasarkan gambar 4. 58 diatas dilakukan perhitungan sebagai berikut :

(1) Menghitung *Cyclomatic Complexcity* $V(G)$ pada *egde* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 6$$

$$N \text{ (node)} = 6$$

$$P \text{ (Predikat node)} = 1$$

Penyelesaian :

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

$$= 6 - 6 + 2$$

$$= 2$$

$$\text{Predikat (P)} = P + 1$$

$$= 1 + 1$$

$$= 2$$

(2) Berdasarkan perhitungan *Cyclomatic Complexcity* dari *Flowgraph* di atas memiliki *Region* = 2

(3) *Independent path* pada *Flowgraph* tersebut yakni:

Path 1 = 1 – 2 – 3 – 4 – 3

Path 2 = 1 – 2 – 3 – 4 – 5 – 6

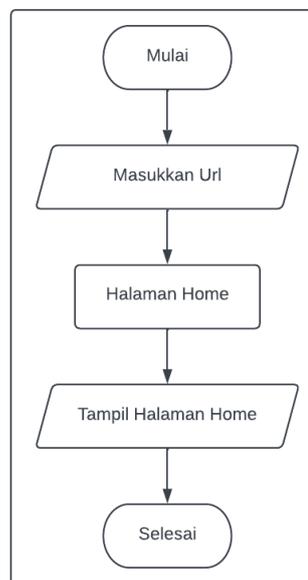
(4) *Grafik matriks* hapus data

Tabel 4. 33 Grafik matriks Hapus Data

	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	
	SUM (E + 1)							$1 + 1 = 2$

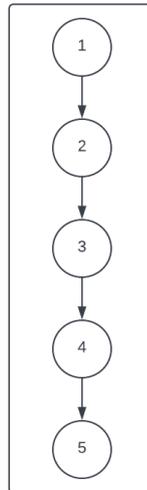
f. *White Box Testing* Halaman *Home Website*

1) *Flowchart*



Gambar 4. 59 *Flowchart* Halaman Home Website

2) *Flowgraph*



Gambar 4. 60 *Flowgraph* Halaman *Home Website*

Berdasarkan gambar 4. 60 diatas dilakukan perhitungan sebagai berikut :

(1) Menghitung *Cyclomatic Complexcity* $V(G)$ pada *egde* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 4$$

$$N \text{ (node)} = 5$$

$$P \text{ (Predikat node)} = 0$$

Penyelesaian :

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

$$= 4 - 5 + 2$$

$$= 1$$

$$\text{Predikat (P)} = P + 1$$

$$= 0 + 1$$

$$= 1$$

(2) Berdasarkan perhitungan *Cyclomatic Complexcity* dari

Flowgraph di atas memiliki *Region* = 1

(3) *Independent path* pada *Flowgraph* tersebut yakni:

Path 1 = 1 – 2 – 3 – 4 – 5

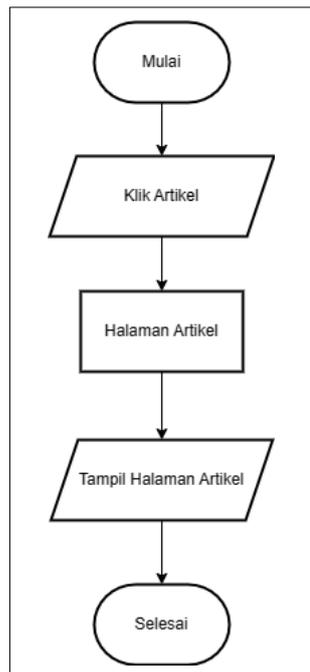
(4) *Grafik matriks* halaman home

Tabel 4. 34 Grafik matriks Halaman Home Website

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

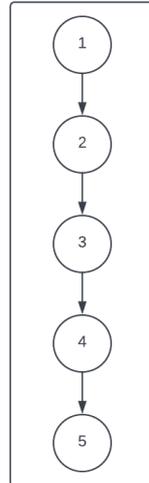
g. *White Box Testing* Halaman Artikel

1) *Flowchart*



Gambar 4. 61 *Flowchart* Halaman Artikel

2) *Flowgraph*



Gambar 4. 62 *Flowgraph* Halaman Artikel

Berdasarkan gambar 4. 62 diatas dilakukan perhitungan sebagai berikut :

(1) Menghitung *Cyclomatic Complexcity* $V(G)$ pada *egde* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 4$$

$$N \text{ (node)} = 5$$

$$P \text{ (Predikat node)} = 0$$

Penyelesaian :

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

$$= 4 - 5 + 2$$

$$= 1$$

$$\text{Predikat (P)} = P + 1$$

$$= 0 + 1$$

$$= 1$$

(2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* di atas memiliki *Region* = 1

(3) *Independent path* pada *Flowgraph* tersebut yakni:

Path 1 = 1 – 2 – 3 – 4 – 5

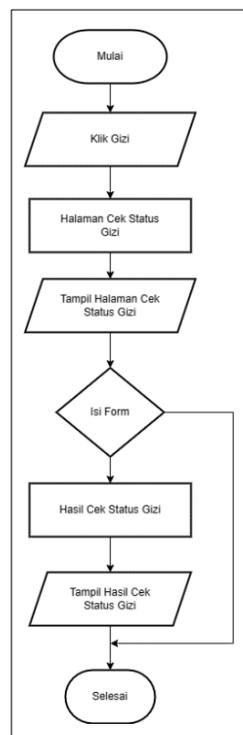
(4) *Grafik matriks* halaman artikel

Tabel 4. 35 Grafik matriks Halaman Artikel

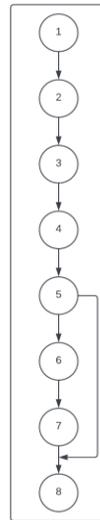
	1	2	3	4	5	E-1
1		1				$1 - 1 = 0$
2			1			$1 - 1 = 0$
3				1		$1 - 1 = 0$
4					1	$1 - 1 = 0$
5						0
	SUM (E + 1)					$0 + 1 = 1$

h. *White Box Testing* Halaman Cek Status Gizi

1) *Flowchart*



Gambar 4. 63 *Flowchart* Cek Status Gizi

2) *Flowgraph*Gambar 4. 64 *Flowgraph* Halaman Cek Status Gizi

Berdasarkan gambar 4. 64 diatas dilakukan perhitungan sebagai berikut :

(1) Menghitung *Cyclomatic Complexity* $V(G)$ pada *egde* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 8$$

$$N \text{ (node)} = 8$$

$$P \text{ (Predikat node)} = 1$$

Penyelesaian :

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

$$= 8 - 8 + 2$$

$$= 2$$

$$\text{Predikat (P)} = P + 1$$

$$= 1 + 1$$

$$= 2$$

(2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* di atas memiliki *Region* = 2

(3) *Independent path* pada *Flowgraph* tersebut yakni:

Path 1 = 1 – 2 – 3 – 4 – 5 – 8

Path 2 = 1 – 2 – 3 – 4 – 5 – 6 – 7 – 8

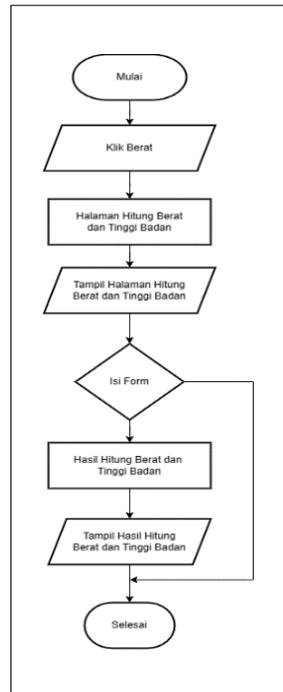
(4) *Grafik matriks* halaman cek status gizi

Tabel 4. 36 *Grafik matriks* Halaman cek status gizi

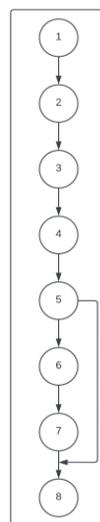
	1	2	3	4	5	6	7	8	E - 1
1		1							$1 - 1 = 0$
2			1						$1 - 1 = 0$
3				1					$1 - 1 = 0$
4					1				$1 - 1 = 0$
5						1		1	$2 - 1 = 1$
6							1		$1 - 1 = 0$
7								1	$1 - 1 = 0$
8									0
	SUM (E + 1)								$1 + 1 = 2$

i. *White Box Testing* Halaman Hitung Berat dan Tinggi Badan

1) *Flowchart*



Gambar 4. 65 *Flowchart* Halaman Hitung Berat dan Tinggi Badan
2) *Flowgraph*



Gambar 4. 66 *Flowgraph* Halaman Hitung Berat dan Tinggi Badan

Berdasarkan gambar 4. 66 diatas dilakukan perhitungan sebagai berikut :

(1) Menghitung *Cyclomatic Complexity* $V(G)$ pada *egde* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E \text{ (edge)} = 8$$

$$N \text{ (node)} = 8$$

$$P \text{ (Predikat node)} = 1$$

Penyelesaian :

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

$$= 8 - 8 + 2$$

$$= 2$$

$$\text{Predikat (P)} = P + 1$$

$$= 1 + 1$$

$$= 2$$

(2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* di atas memiliki *Region* = 2

(3) *Independent path* pada *Flowgraph* tersebut yakni:

$$\text{Path 1} = 1 - 2 - 3 - 4 - 5 - 8$$

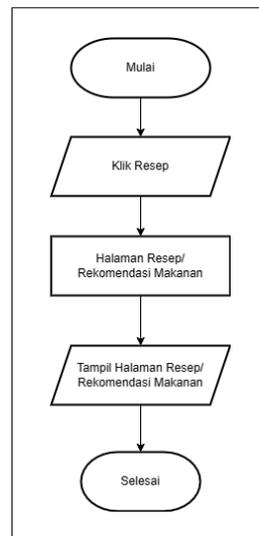
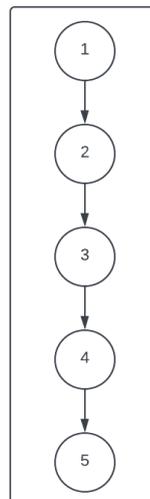
$$\text{Path 2} = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8$$

(4) *Grafik matriks* halaman hitung berat dan tinggi badan

Tabel 4. 37 Grafik matriks Hitung Berat dan Tinggi Badan

	1	2	3	4	5	6	7	8	E - 1
1		1							1 - 1 = 0
2			1						1 - 1 = 0
3				1					1 - 1 = 0
4					1				1 - 1 = 0
5						1		1	2 - 1 = 1
6							1		1 - 1 = 0
7								1	1 - 1 = 0
8									0
	SUM (E + 1)								1 + 1 = 2

j. *White Box Testing* Halaman Resep/Rekomendasi Makanan

1) *Flowchart*Gambar 4. 67 *Flowchart* Halaman Resep/Rekomendasi Makanan2) *Flowgraph*Gambar 4. 68 *Flowgraph* Halaman Resep/Rekomendasi Makanan

Berdasarkan gambar 4. 68 diatas dilakukan perhitungan sebagai berikut :

(1) Menghitung *Cyclomatic Complexcity* $V(G)$ pada *egde* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E (\text{edge}) = 4$$

$$N (\text{node}) = 5$$

$$P (\text{Predikat node}) = 0$$

Penyelesaian :

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

$$= 4 - 5 + 2$$

$$= 1$$

$$\text{Predikat (P)} = P + 1$$

$$= 0 + 1$$

$$= 1$$

(2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* di atas memiliki *Region* = 1

(3) *Independent path* pada *Flowgraph* tersebut yakni:

$$\text{Path 1} = 1 - 2 - 3 - 4 - 5$$

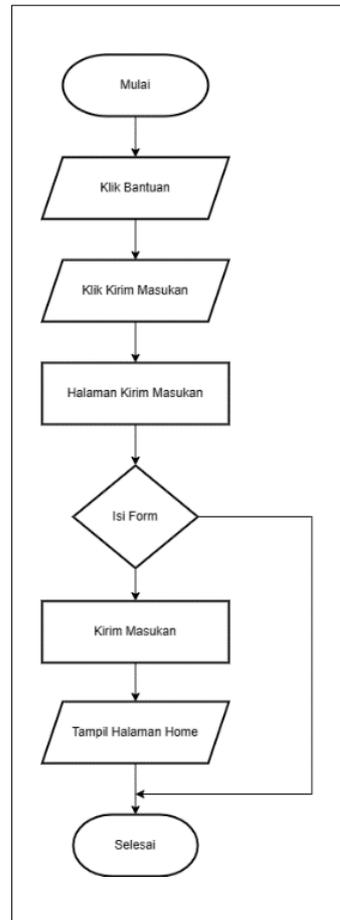
(4) *Grafik matriks* halaman resep/rekomendasi makanan

Tabel 4. 38 Grafik matriks Resep/Rekomendasi Makanan

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

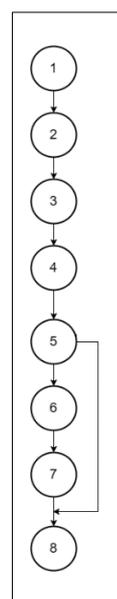
k. *White Box Testing* Halaman Kirim Masukan

1) *Flowchart*



Gambar 4. 69 *Flowchart* Kirim Masukan

2) *Flowgraph*



Gambar 4. 70 *Flowgraph* Kirim Masukan

Berdasarkan gambar 4. 70 diatas dilakukan perhitungan sebagai berikut :

(1) Menghitung *Cyclomatic Complexity* $V(G)$ pada *egde* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E (\text{edge}) = 8$$

$$N (\text{node}) = 8$$

$$P (\text{Predikat node}) = 1$$

Penyelesaian :

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

$$= 8 - 8 + 2$$

$$= 2$$

$$\text{Predikat (P)} = P + 1$$

$$= 1 + 1$$

$$= 2$$

(2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* di atas memiliki *Region* = 2

(3) *Independent path* pada *Flowgraph* tersebut yakni:

$$\text{Path 1} = 1 - 2 - 3 - 4 - 5 - 8$$

$$\text{Path 2} = 1 - 2 - 3 - 4 - 5 - 6 - 7 - 8$$

(4) *Grafik matriks* cari kirim masukan

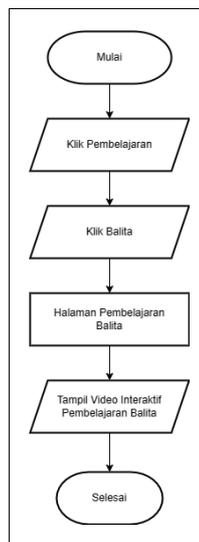
Tabel 4. 39 Grafik matriks Kirim Masukan

	1	2	3	4	5	6	7	8	E - 1
1		1							$1 - 1 = 0$
2			1						$1 - 1 = 0$
3				1					$1 - 1 = 0$
4					1				$1 - 1 = 0$

5						1		1	$2 - 1 = 1$
6							1		$1 - 1 = 0$
7								1	$1 - 1 = 0$
8									0
SUM (E + 1)									$1 + 1 = 2$

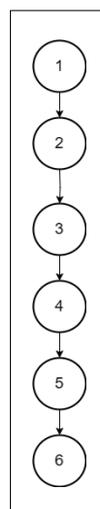
1. *White Box Testing* Halaman Pembelajaran Balita

1) *Flowchart*



Gambar 4. 71 *Flowchart* Halaman Pembelajaran Balita

2) *Flowgraph*



Gambar 4. 72 *Flowgraph* Halaman Pembelajaran Balita

Berdasarkan gambar 4. 72 diatas dilakukan perhitungan sebagai berikut :

(1) Menghitung *Cyclomatic Complexity* $V(G)$ pada *egde* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E (\text{edge}) = 5$$

$$N (\text{node}) = 6$$

$$P (\text{Predikat node}) = 0$$

Penyelesaian :

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

$$= 5 - 6 + 2$$

$$= 1$$

$$\text{Predikat (P)} = P + 1$$

$$= 0 + 1$$

$$= 1$$

(2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* di atas memiliki *Region* = 1

(3) *Independent path* pada *Flowgraph* tersebut yakni:

$$\text{Path 1} = 1 - 2 - 3 - 4 - 5 - 6$$

(4) *Grafik matriks* halaman pembelajaran balita

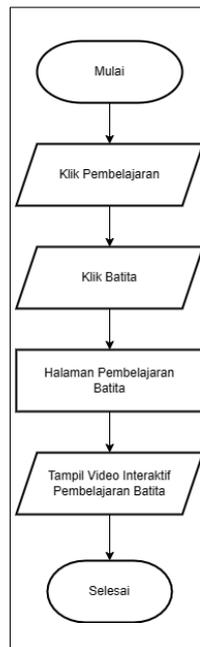
Tabel 4. 40 Grafik Halaman Pembelajaran Balita

	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 - 1 = 0
5						1	1 - 1 = 0
6							0

	SUM (E + 1)		0 + 1 = 1
--	-------------	--	-----------

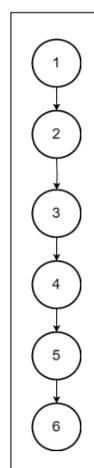
m. *White Box Testing* Halaman Pembelajaran Batita

1) *Flowchart*



Gambar 4. 73 *Flowchart* Pembelajaran Batita

2) *Flowgraph*



Gambar 4. 74 *Flowgraph* Pembelajaran Batita

Berdasarkan gambar 4. 74 diatas dilakukan perhitungan sebagai

berikut :

(1) Menghitung *Cyclomatic Complexity* $V(G)$ pada *egde* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E (\text{edge}) = 5$$

$$N (\text{node}) = 6$$

$$P (\text{Predikat node}) = 0$$

Penyelesaian :

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

$$= 5 - 6 + 2$$

$$= 1$$

$$\text{Predikat (P)} = P + 1$$

$$= 0 + 1$$

$$= 1$$

(2) Berdasarkan perhitungan *Cyclomatic Complexity* dari *Flowgraph* di atas memiliki *Region* = 1

(3) *Independent path* pada *Flowgraph* tersebut yakni:

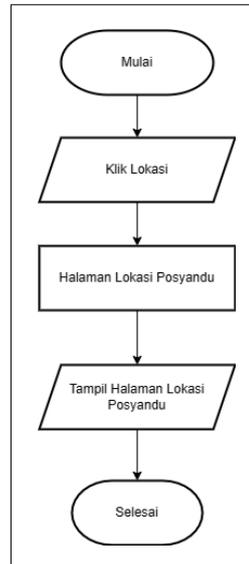
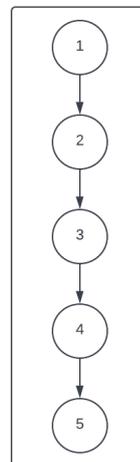
$$\text{Path 1} = 1 - 2 - 3 - 4 - 5 - 6$$

(4) *Grafik matriks* unduh *datasheet*

Tabel 4. 41 Grafik matriks Halaman Pembelajaran Batita

	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 - 1 = 0	
5						1	1 - 1 = 0	
6							0	
	SUM (E + 1)							0 + 1 = 1

n. *White Box Testing* Halaman Lokasi Posyandu

1) *Flowchart*Gambar 4. 75 *Flowchart* Lokasi Posyandu2) *Flowgraph*Gambar 4. 76 *Flowgraph* Lokasi Posyandu

Berdasarkan gambar 4. 76 diatas dilakukan perhitungan sebagai berikut :

(5) Menghitung *Cyclomatic Complexcity* $V(G)$ pada *egde* dan *node*

$$\text{Pada rumus : } V(G) = E - N + 2$$

$$E (\text{edge}) = 4$$

$$N (\text{node}) = 5$$

$$P (\text{Predikat node}) = 0$$

Penyelesaian :

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

$$= 4 - 5 + 2$$

$$= 1$$

$$\text{Predikat (P)} = P + 1$$

$$= 0 + 1$$

$$= 1$$

(6) Berdasarkan perhitungan *Cyclomatic Complexity* dari

Flowgraph di atas memiliki *Region* = 1

(7) *Independent path* pada *Flowgraph* tersebut yakni:

Path 1 = 1 - 2 - 3 - 4 - 5

(8) *Grafik matriks* unduh *datasheet*

Tabel 4. 42 Grafik matriks Halaman Lokasi Posyandu

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

E. Status Gizi Balita

Tumbuh kembang si buah hati sangat bergantung pada berbagai faktor, salah satunya gizi yang dikonsumsi oleh anak. Kekurangan asupan gizi dalam 1000 hari pertama kehidupan anak akan menyebabkan berbagai masalah kesehatan, seperti stunting (tubuh pendek) dan wasting (tubuh terlalu kurus). Berikut adalah tabel gizi balita berdasarkan usia, berat badan, dan tinggi badan menurut Edwin Pranata (2011).

Tabel 4. 43 Status Gizi Balita

No	Usia (Bln)	Berat (Kg)	Tinggi (cm)	Status Gizi
1	12	10	70	Gizi Baik
2	13	11	71	Gizi Baik
3	14	12	72	Gizi Baik
4	15	13	73	Kelebihan Gizi
5	16	14	74	Kelebihan Gizi
6	17	15	75	Kelebihan Gizi
7	18	16	76	Kelebihan Gizi
8	19	17	77	Kelebihan Gizi
9	20	18	78	Kelebihan Gizi
10	21	19	79	Kelebihan Gizi
11	22	20	80	Kelebihan Gizi
12	23	21	81	Kelebihan Gizi
13	24	22	82	Kelebihan Gizi
14	25	23	83	Kelebihan Gizi
15	26	24	84	Kelebihan Gizi

BAB V

PENUTUP

A. Kesimpulan

Dalam penelitian ini, telah dibahas bagaimana pengembangan aplikasi multimedia interaktif yang bertujuan untuk membantu orang tua dalam mengawasi pertumbuhan dan perkembangan anak usia balita dan batita. Aplikasi ini menerapkan teknologi modern dengan menggunakan *framework Laravel* sebagai bahasa pemrograman utama, *Visual Studio Code (VSCode)* sebagai alat pengembangan, dan *MySQL* sebagai sistem basis data. Hasil dari penelitian ini adalah suatu platform web multimedia interaktif yang menyediakan beragam video pembelajaran yang sangat berguna bagi orang tua untuk memahami lebih baik tentang perkembangan anak-anak mereka. Selain itu, aplikasi ini juga dilengkapi dengan fitur pengecekan status gizi dan perhitungan berat dan tinggi badan anak, yang sangat berguna untuk memantau kesehatan anak secara berkala.

B. Saran

Dalam penelitian ini, penulis menyadari adanya beberapa kekurangan yang memerlukan perbaikan dan pengembangan di penelitian berikutnya. Maka dari itu, penulis ingin memberikan beberapa saran untuk perkembangan selanjutnya, sebagai berikut:

1. Mengembangkan aplikasi pada beragam *platform*, seperti *iOS* dan *Android*, untuk dapat menjangkau lebih banyak pengguna.

2. Aplikasi ini dapat dilakukan pembaruan pada tampilan antarmuka untuk membuatnya lebih menarik dan sesuai dengan tren terkini
3. Guna meningkatkan kegunaan aplikasi ini, disarankan untuk melanjutkan pengembangannya. Contohnya, mencakup penambahan fitur personalisasi, seperti profil anak, sehingga aplikasi dapat memberikan rekomendasi yang lebih tepat sesuai dengan perkembangan masing-masing anak

DAFTAR PUSTAKA

- Abdul Kahfi Assidiq, (2020) dalam kamus Biologi, laboratorium, Jakarta: MediaKompitindo
- Adi nugroho (2019), “*Unified Modelling Language* Jakarta: MediaKompitindo
- Alan Nur Aditya, 2019, Internet Protocol (IP) dan Transmission Control Protocol (TCP)
- Alexander F. K, Sibero. 2019. Kitab Suci Web Progremining. MediaKom.Yogyakarta. Bahra, Al. 2019. Analisis dan Desain Sistem Informasi. Yogyakarta: Graha Ilmu.
- Dio Arisyahputra (2020), Program Studi Sistem Komputer, Fakultas Teknik dan Ilmu Komputer, **Universitas** Komputer Indonesia dalam penelitiannya yang berjudul “**Perancangan Aplikasi Pemantauan Tumbuh Kembang Anak Berbasis Android**”
- Hartono (2021) aplikasi penggunaan dalam suatu komputer, instruksi (*instruction*) Jakarta: Draha Ilmu
- HELMI LUTFIAN (2021), Program Studi Teknik Informatika Fakultas Komunikasi Dan Informatika Universitas Muhammadiyah Surakarta 2022, dalam penelitiannya yang berjudul “**SISTEM INFORMASI POSYANDU BERBASIS WEB**”
- Jogiyanto, H.M. 2020. Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur teori dan Praktik Aplikasi Bisnis. Andi. Yogyakarta.
- Madcoms Madiun, 2019, HTML (*Hyper Text Markup Language*) editor profesional, Jakarta: Raja GrafindoPersada,
- Madiun, 2020, Multimedia editor profesional, Jakarta: Raja Grafindo Persada,
- Merita. (2019) “TUMBUH KEMBANG ANAK USIA 0-5 TAHUN”
https://scholar.google.co.id/scholar?hl=id&as_sdt=0%2C5&q=jurnal+abdimas+kesehatan+tumbuh+kembang+anak&btnG=#d=gs_qabs&t=1703759937217&u=%23p%3DdpQ20vL5NLUJ, diakses 24 desember 2023

- Nugroho, Bunafit. 2020. *Pemrograman Web: Membuat Sistem Informasi Akademik Sekolah dengan PHP-MySql dan Dreamweaver*. Yogyakarta: Gava Media.
- Ramadhan Arif. 2021. *Pemrograman Web Database dengan PHP dan MySQL*. Jakarta: PT Elex Media Komputindo.
- Presman, Roger.S. 2019. *Rekayasa Perangkat Lunak*. Andi. Yogyakarta.
- Santoso, Harip. 2019. *Aplikasi Web/asp.net + cd*. Jakart: Elex MediaKompitindo.
- Sukamto, Rosa Ariani. 2019. *Analisa dan Desain Sistem Informasi*. Bandung: Informatika.
- Edwin Pranata.(2011). *engenal Status Gizi Anak dan Cara Menilainya*. Diakses pada 19 agustus 2024 dari <https://realfood.co.id/id/artikel/mengenal-status-gizi-anak-dan-cara-menilainya>.