

BAB 3 METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Dalam melaksanakan penelitian ini, terdapat tahapan-tahapan yang harus dilakukan dimulai dari studi literatur, pengumpulan data, perancangan sistem, implementasi, dan pengujian. Berikut merupakan penjelasan setiap tahapan.

A. Studi Literatur

Untuk dapat memahami lebih dalam mengenai topik permasalahan, dilakukan studi literatur yang terbagi dalam 3 jenis kegiatan. Pertama, konsultasi langsung dengan dr. Ivony Yuyu Susanto, Sp.A yang bertugas di rumah sakit St. Carolus untuk mendapatkan pemahaman lebih dalam mengenai penyakit *stunting* pada anak. Kedua, melakukan eksplorasi terhadap studi pustaka dan artikel-artikel kesehatan terkait kondisi gizi anak dan algoritma sistem pakar yang akan diimplementasikan. Ketiga, membuat *study literature review* (SLR) berdasarkan penelitian terdahulu mengenai *stunting* dan sistem pakar.

B. Pengumpulan Data

Pengumpulan data dilakukan dengan melakukan kunjungan ke Puskesmas Batuceper untuk berdiskusi dan wawancara secara langsung dengan Dian Rahmawati, S.Gz. Data yang diperoleh merupakan hasil rekomendasi nutrisi atau gizi untuk setiap interval usia dengan berbagai kondisi. Data akan diolah menjadi aturan-aturan sistem pakar dalam penelitian ini.

C. Perancangan Sistem

Sistem dirancang berdasarkan kebutuhan dan informasi yang telah diperoleh sebelumnya untuk disusun menjadi sebuah sistem yang berfungsi dengan baik melalui pembuatan *unified modelling language* (UML) seperti *use case*, *activity diagram*, *sequence diagram*, dan *class diagram*. Perancangan dilanjutkan dengan membuat struktur basis data dan juga rancangan tampilan antarmuka.

D. Implementasi

Hasil perancangan sistem akan diimplementasikan ke dalam pembangunan aplikasi android. Dalam implementasi ini, bahasa pemrograman yang digunakan adalah Java.

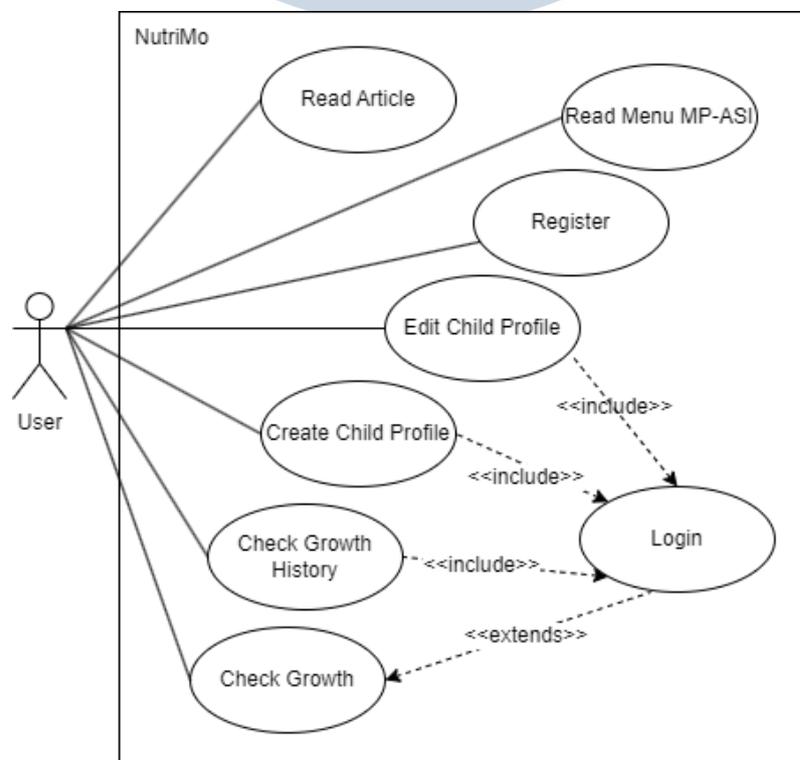
E. Pengujian

Pada proses pengujian aplikasi akan dilakukan dalam 2 jenis pengujian, yaitu pengujian fitur dalam sistem dengan metode *black box* dan pengujian akurasi dari hasil yang diberikan oleh sistem yang akan dilakukan oleh Dian Rahmawati, S.Gz. Kemudian, aplikasi akan disebarakan melalui Google Drive untuk dicoba oleh para responden yang akan mengisi angket pertanyaan untuk mengukur tingkat kepuasan dengan metode EUCS dan skala Likert.

3.2 Perancangan Sistem

Perancangan ini bertujuan untuk menjelaskan alur sistem secara keseluruhan. Pada tahap ini, perancangan dimulai dengan membuat *use case*, *activity diagram*, *sequence diagram*, *class diagram*, struktur *database*, hingga tampilan antarmuka.

3.2.1 Use Case Diagram



Gambar 3.1. Use Case Diagram

Gambar 3.1 menjelaskan hubungan antara aktor dengan kegiatan *use case*. Kegiatan yang dapat dilakukan oleh aktor dijelaskan pada Tabel 3.1.

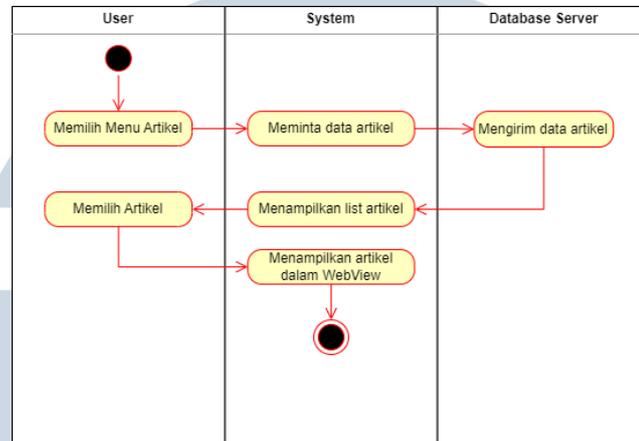
Tabel 3.1. Tabel Kegiatan *Use Case*

No	Aktor	Use Case	Keterangan
1	<i>User</i>	<i>Read Article</i>	<i>User</i> dapat membaca artikel yang disediakan.
2	<i>User</i>	<i>Read Menu MP-ASI</i>	<i>User</i> dapat membaca resep menu MP-ASI yang disediakan
3	<i>User</i>	<i>Login</i>	<i>User</i> dapat masuk ke dalam aplikasi dengan akun yang telah dimiliki.
4	<i>User</i>	<i>Register</i>	<i>User</i> dapat membuat akun untuk aplikasi dengan <i>email</i> .
5	<i>User</i>	<i>Check Growth History</i>	<i>User</i> dapat melihat kembali detail-detail hasil pemeriksaan yang sudah dilakukan.
6	<i>User</i>	<i>Create Child Profile</i>	<i>User</i> dapat menambah atau membuat profil anak untuk memantau tumbuh kembang anak
7	<i>User</i>	<i>Edit Child Profile</i>	<i>User</i> dapat merubah data dari profil anak yang sudah dibuat sebelumnya
8	<i>User</i>	<i>Check Growth</i>	<i>User</i> dapat melakukan diagnosa kondisi <i>stunting</i> dan gizi serta mendapatkan rekomendasi gizi.

3.2.2 Activity Diagram

Untuk menggambarkan alur sistem, keputusan yang mungkin terjadi, hingga bagaimana aktivitas berakhir dapat menggunakan *activity diagram*. Dalam pengembangan aplikasi NutriMo, berikut 8 buah *activity diagram* yang telah dibentuk berdasarkan *use case*.

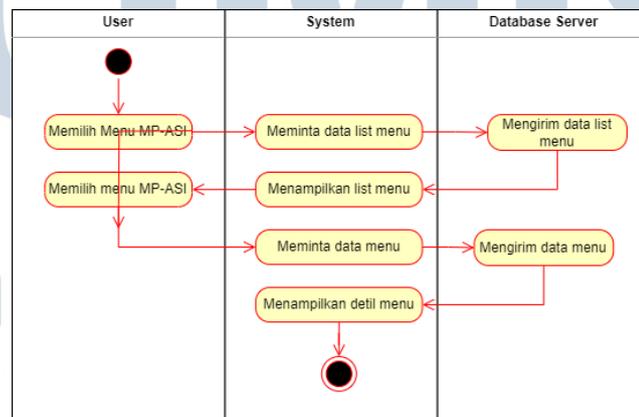
A Read Article



Gambar 3.2. Activity diagram Read Article

Untuk membaca artikel yang telah disediakan, *user* dapat memilih menu Artikel. Setelah itu, sistem akan menampilkan daftar artikel yang tersedia pada *database*. *User* dapat melanjutkan dengan memilih artikel yang ingin dibuka. Ketika *user* telah menentukan, sistem akan menampilkan detail artikel pada tampilan aplikasi.

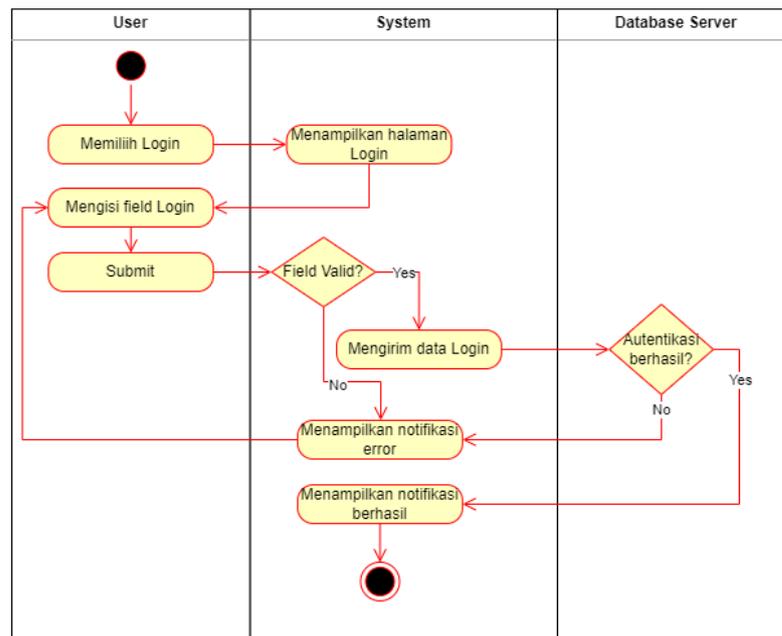
B Read Menu MP-ASI



Gambar 3.3. Activity diagram Read Menu MP-ASI

Untuk mengakses menu-menu MP-ASI yang tersedia, *user* dapat membuka Menu terlebih dahulu. Kemudian sistem akan menampilkan daftar menu MP-ASI yang telah diperoleh dari *database*. Saat *user* memilih menu yang ingin dibaca, sistem akan menampilkan detail menu tersebut dari *database*.

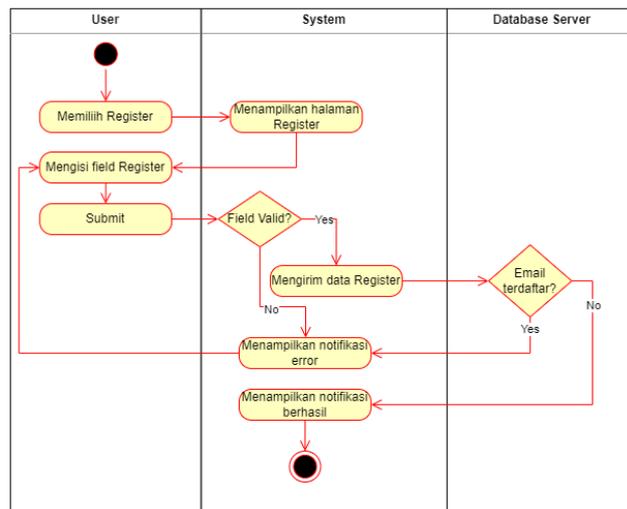
C Login



Gambar 3.4. Activity diagram Login

Pada aplikasi ini, *user* dapat melakukan *login* dengan akun yang telah dimiliki. Proses untuk *login* dimulai saat *user* membuka halaman *login*, *user* akan ditampilkan formulir untuk melakukan *login* yang terdiri dari *email* dan kata sandi. Setelah *user* mengisi dan menekan tombol "Login", sistem akan memeriksa apakah formulir sudah terisi semua, jika sudah sistem akan lanjutkannya ke *database* untuk autentikasi data pengguna. Jika berhasil, *user* akan berhasil *login* dan kembali ke halaman utama. Namun, jika terdapat tahapan dimana terdapat *error* atau kesalahan, sistem akan menampilkan notifikasi yang menampilkan permasalahan yang ada.

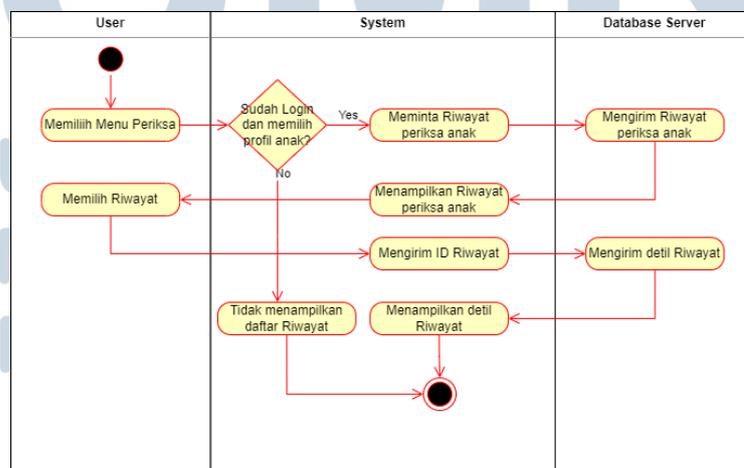
D Register



Gambar 3.5. Activity diagram Register

Bagi *user* yang belum memiliki akun, dapat mendaftarkan akun pada menu Registrasi. Ketika *user* mengakses halaman registrasi, akan disuguhkan dengan formulir untuk mendaftarkan akun dengan mengisi *email*, kata sandi, dan konfirmasi sandi lalu menekan tombol "Register". Sistem akan melanjutkan untuk melakukan validasi input dan validasi *email*. Ketika validasi berhasil, *user* akan secara otomatis kembali ke halaman utama dengan akun yang baru didaftarkan.

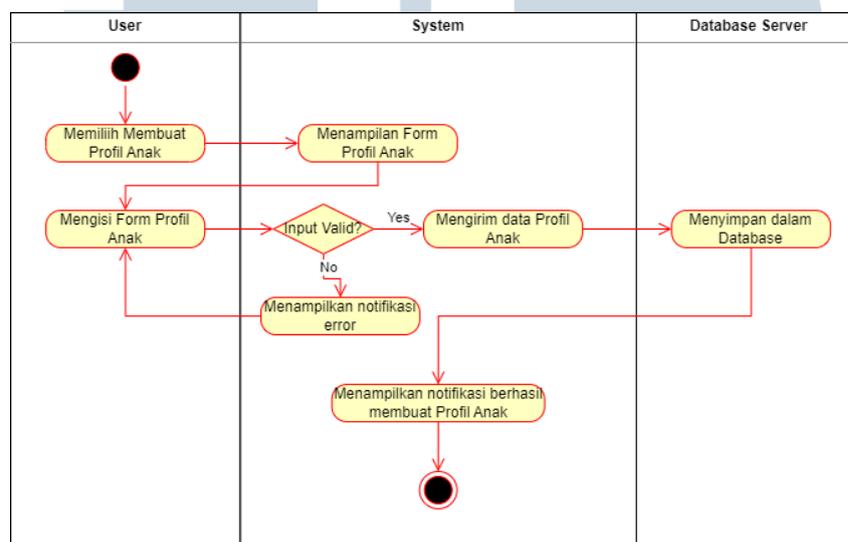
E Check Growth History



Gambar 3.6. Activity diagram Check Growth History

Proses melihat riwayat periksa diawali dengan memilih fitur Periksa. Setelah itu sistem akan memeriksa akun dan profil anak yang sedang aktif. Kemudian sistem meminta data semua riwayat milik profil anak yang aktif. *User* dapat memilih riwayat periksa yang telah ditampilkan. Kemudian sistem akan meminta detail riwayat pada *database*, lalu sistem akan menampilkan data tersebut.

F Create Child Profile

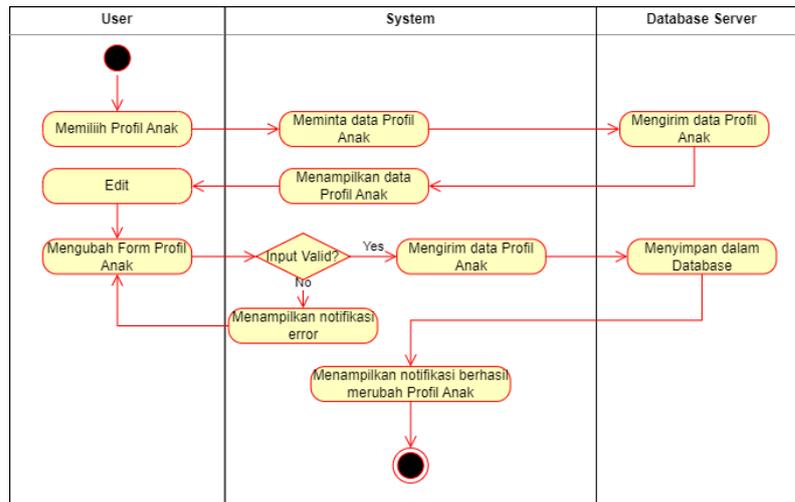


Gambar 3.7. Activity diagram Create Child Profile

Untuk menambah profil anak, *user* dapat menekan tombol "Tambah Profil Anak" untuk mengkases formulir profil anak. Kemudian *user* dapat mengisi formulir tersebut dan mengirimnya. Sistem akan memeriksa input dari *user* kemudian menyimpannya dalam *database*. Ketika berhasil, *user* akan menerima notifikasi bahwa profil anak berhasil terdaftar.

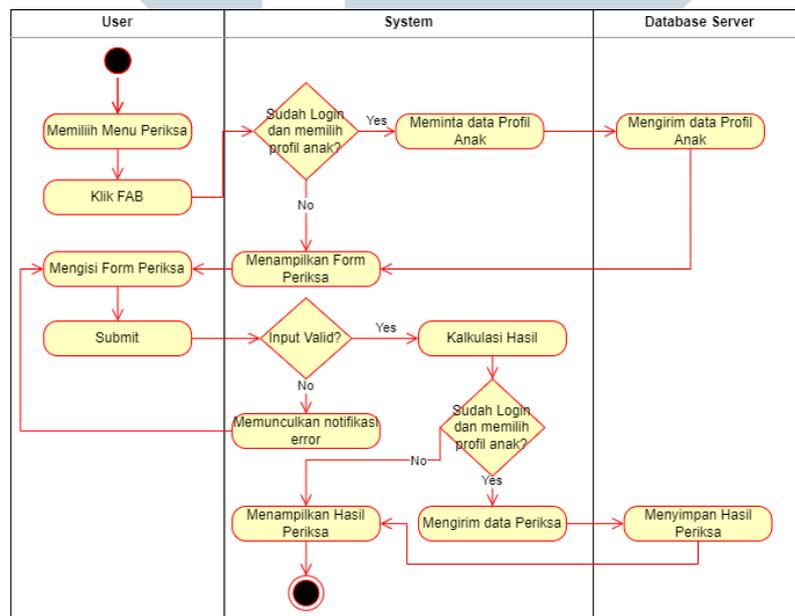
G Edit Child Profile

Setelah membuat profil anak, *user* dapat memberikan perubahan pada profil anak dengan memilih profil anak yang ingin dirubah. Kemudian sistem akan mengambil data profil anak tersebut sebagai profil aktif. Lalu *user* dapat mengkases formulir ubah profil anak pada pengaturan. Setelah merubah data profil anak, *user* diharuskan menekan tombol "Simpan" agar sistem dapat mengirim ke *database* untuk disimpan.



Gambar 3.8. Activity diagram Edit Child Profile

H Check Growth



Gambar 3.9. Activity diagram Check Growth

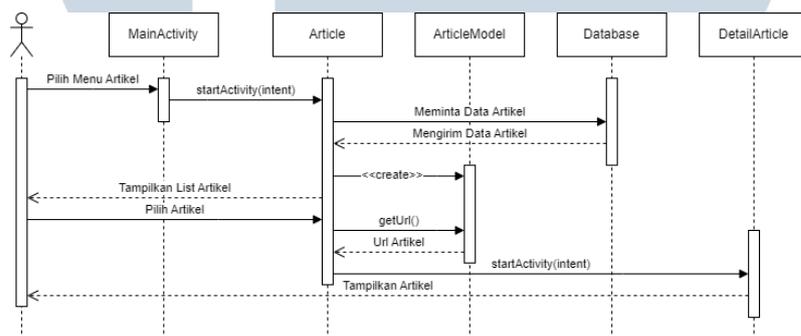
Untuk melakukan pemeriksaan pada anak, proses diawali dengan *user* membuka menu Periksa, lalu sistem akan memeriksa akun dan profil anak yang dipilih. Kemudian *user* dapat menekan *float action button* atau tanda tambah untuk membuka formulir periksa. Setelah *user* mengisi formulir tersebut, sistem akan melakukan validasi terhadap input yang diberikan. Jika tidak ditemukan kesalahan,

sistem akan melakukan proses kalkulasi. Hasil pemeriksaan akan disimpan oleh sistem pada *database* bila *user* telah melakukan *login* dan memilih profil anak. Jika *user* belum melakukan hal tersebut, hasil periksa tidak akan tersimpan.

3.2.3 Sequence Diagram

Dalam menggambarkan bagaimana antar objek berinteraksi secara urut waktu (vertikal) dan objek yang terkait (horizontal) dapat menggunakan *sequence diagram*. Berdasarkan *use case* dan *activity diagram* yang telah dibentuk sebelumnya, berikut merupakan *sequence diagram* yang terbentuk.

A Read Article

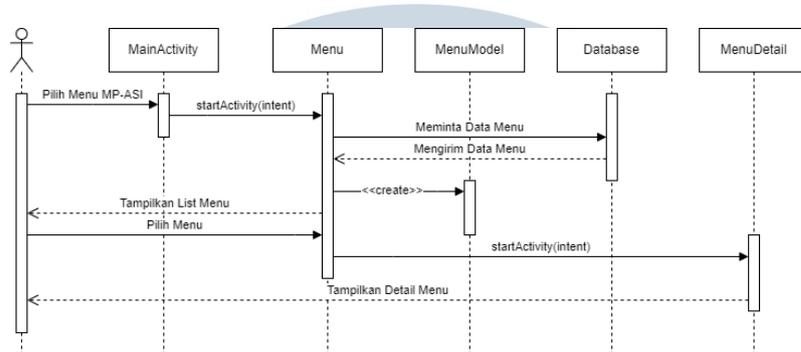


Gambar 3.10. *Sequence diagram Read Article*

Untuk menampilkan artikel yang ingin dibaca oleh *user*. Proses diawali dengan memilih menu Artikel dari halaman utama. Sistem akan membuka *activity* bernama *Article* yang meminta data informasi artikel kepada *database* yang dibentuk sebagai objek dari *ArticleModel*. Ketika *user* memilih salah satu artikel, sistem akan membuka *activity* *DetailArticle* dengan mengirim data berupa URL dari artikel tersebut untuk dibuka dengan *WebView*.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

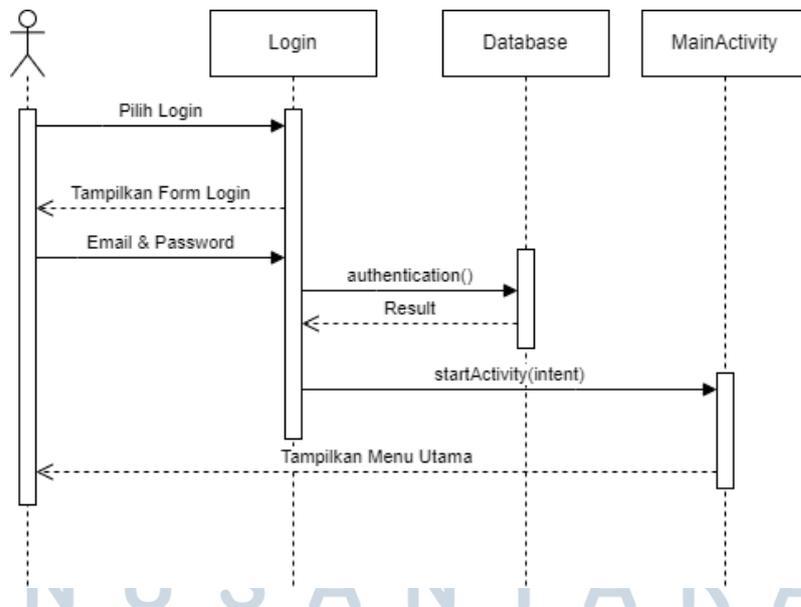
B Read Menu MP-ASI



Gambar 3.11. *Sequence diagram Read Menu MP-ASI*

Untuk menampilkan resep menu MP-ASI, *user* dapat mengakes Menu dari halaman utama. Ketika *activity* Menu dijalankan, sistem akan meminta data menu ke *database* yang dibentuk menjadi objek dari *MenuModel*. Saat *user* memilih salah satu menu MP-ASI, objek *MenuModel* akan dikirimkan ke *activity* *MenuDetail* untuk ditampilkan secara detail.

C Login

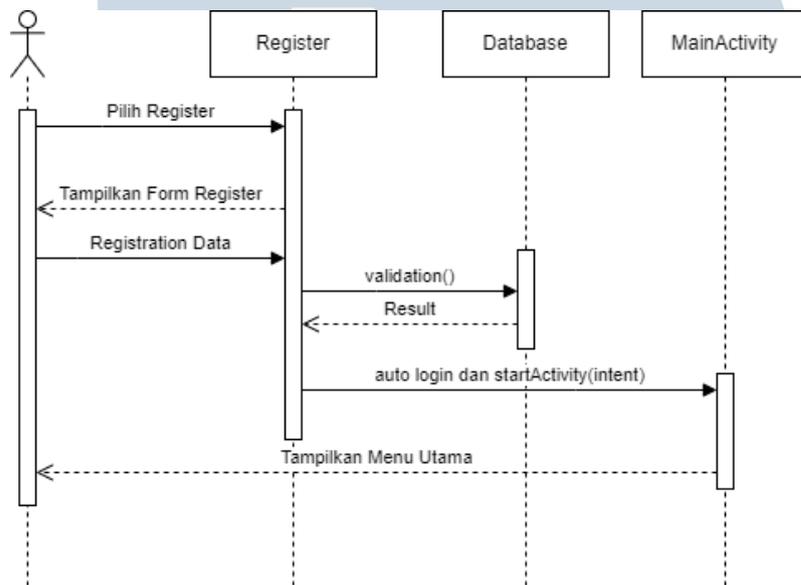


Gambar 3.12. *Sequence diagram Login*

Untuk proses *login*, diawali dengan *user* membuka halaman *Login*. *User* akan ditampilkan formulir untuk melakukan *login* dengan *email* dan kata sandi.

Setelah *user* mengisi dan menekan tombol "Login", sistem akan menjalankan fungsi *authentication()* untuk melakukan validasi input dan autentikasi data pengguna pada *database*. Saat fungsi ini selesai dan hasil yang diberikan merupakan sukses, maka *user* akan secara otomatis kembali ke halaman utama dengan kondisi berhasil *login*.

D Register

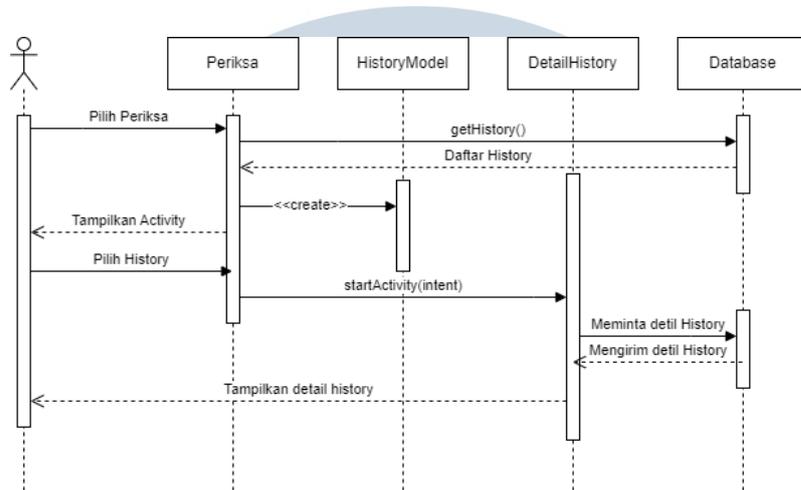


Gambar 3.13. *Sequence diagram Register*

Proses berjalan ketika *user* membuka halaman *Register*. Pada halaman ini akan tersedia formulir yang berisi *email*, kata sandi, dan konfirmasi kata sandi dan tombol "Register". Ketika formulir sudah diisi, sistem akan menjalankan fungsi *validation()* untuk validasi input dan pengecekan terhadap email yang didaftarkan. Ketika berhasil, sistem akan memberikan notifikasi dan mengarahkan kembali ke halaman utama dengan kondisi otomatis *login* dengan akun yang baru didaftarkan.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

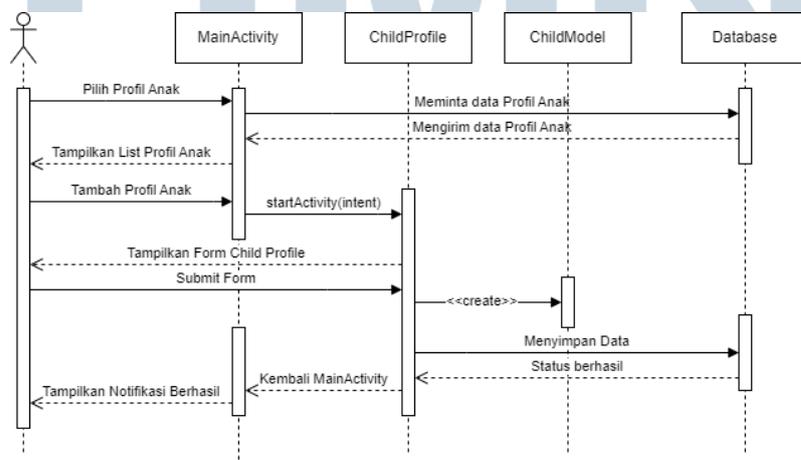
E Check Growth History



Gambar 3.14. Sequence diagram Check Growth History

Untuk membaca kembali detail hasil pemeriksaan, proses diawali dengan mengakses menu Periksa. Ketika menu Periksa diakses, sistem akan meminta data riwayat pemeriksaan berdasarkan profil anak yang sedang aktif atau dipilih ke *database*. Data ini akan disimpan dalam objek dari *HistoryModel*. Ketika *user* memilih salah satu riwayat pemeriksaan, sistem akan membuka *activity DetailHistory* dengan mengirimkan data riwayat untuk dibuka secara detail.

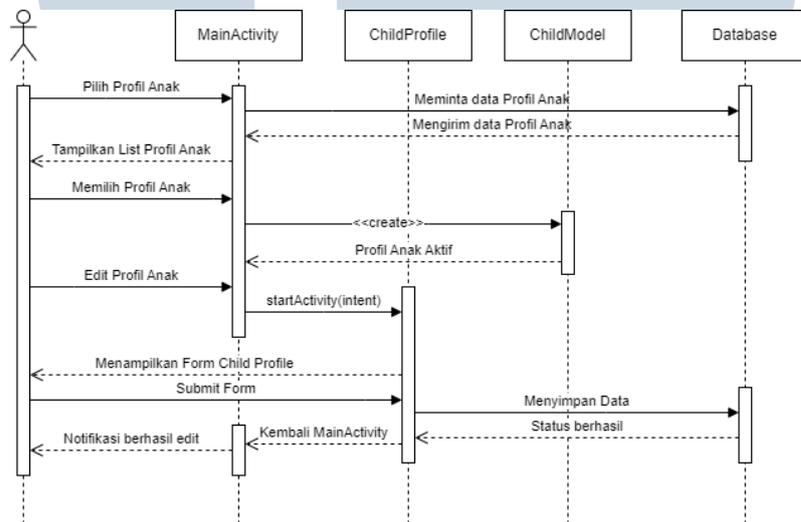
F Create Child Profile



Gambar 3.15. Sequence diagram Create Child Profile

Untuk membuat profil anak, *user* dapat mengakses *list* profil anak pada halaman utama. *List* profil anak merupakan kumpulan profil anak yang didaftarkan oleh akun *user*. Dengan memencet tombol "Tambah Profil Anak", sistem akan membuka *activity ChildProfile*. Pada *activity* ini, sistem menampilkan formulir untuk menambah profil anak. Setelah *user* mengisi dan menekan tombol "Simpan", sistem akan mengirim perintah untuk menyimpan data profil anak tersebut kedalam *database*. Ketika proses ini berhasil, sistem akan kembali ke halaman utama dan menampilkan notifikasi berhasil menambahkan profil anak.

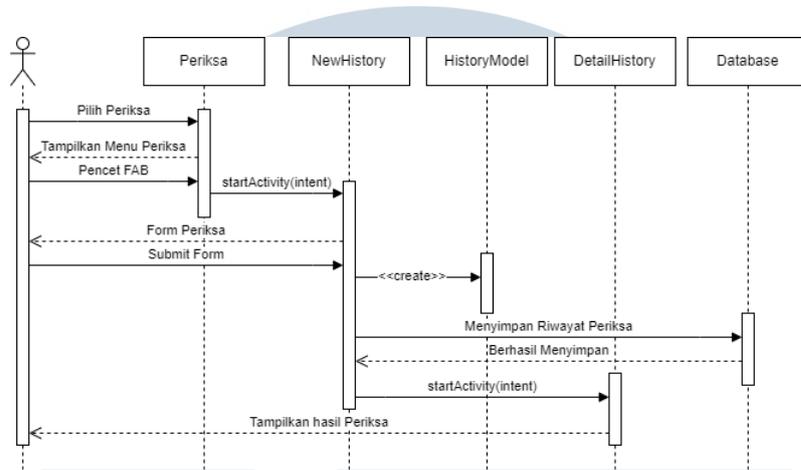
G Edit Child Profile



Gambar 3.16. *Sequence diagram Edit Child Profile*

Profil anak yang sudah terdaftar dapat dilakukan pengeditan. Untuk itu, proses diawali pada halaman utama untuk memilih profil anak yang aktif. Sistem akan membentuk objek profil anak aktif dari dari kelas *ChildModel*. Ketika terdapat profil anak aktif, *user* dapat mengakses halaman edit profil anak. Sistem akan menampilkannya dari *activity ChildProfile* dan menampilkan formulir edit profil anak. Setelah *user* melakukan perubahan data dan memencet tombol "Simpan", sistem akan melakukan validasi input dan mengirim perintah untuk melakukan *update* pada data profil anak tersebut. Ketika proses ini berhasil, sistem akan kembali ke halaman utama dan menampilkan notifikasi berhasil memperbaharui profil anak.

H Check Growth

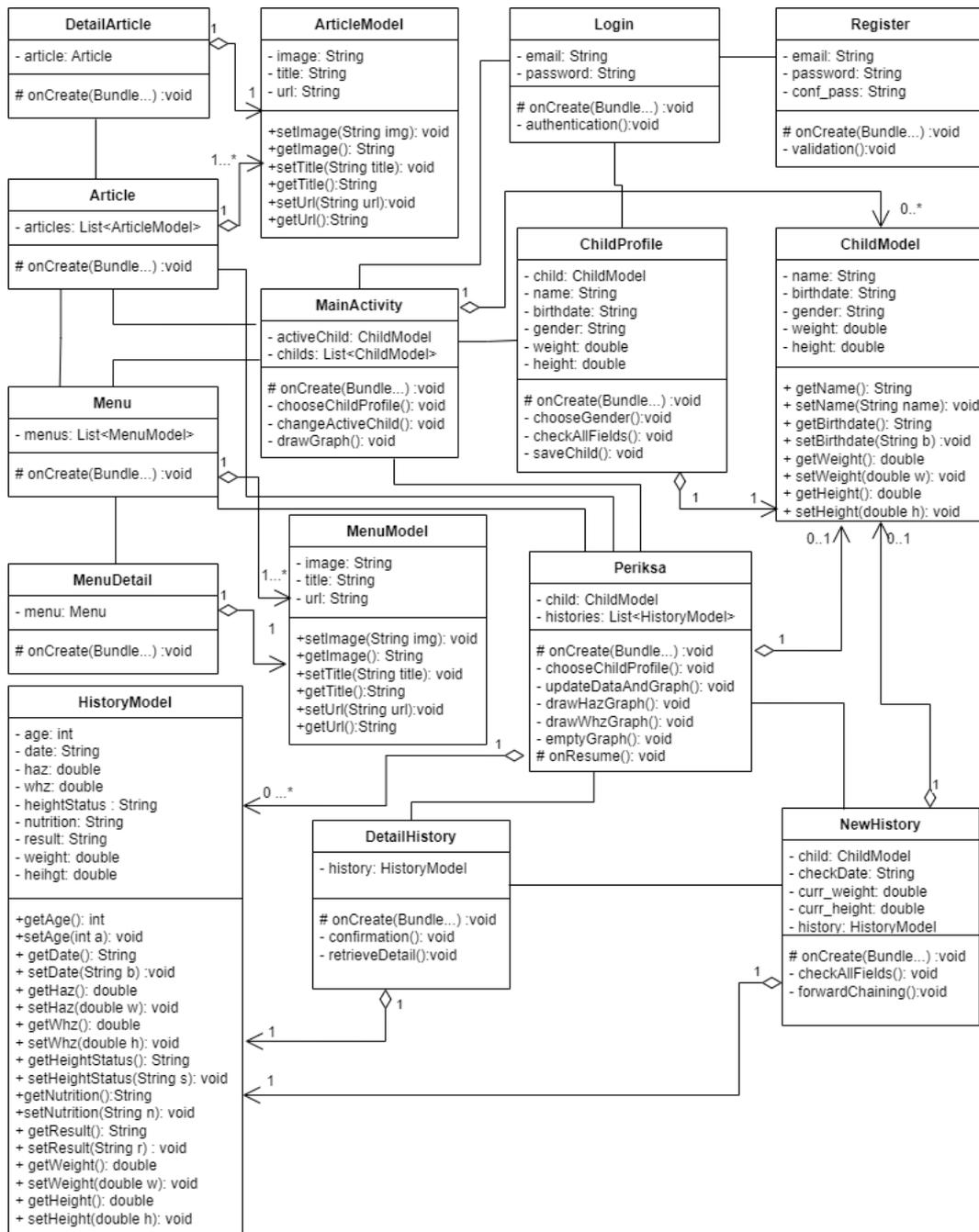


Gambar 3.17. Sequence diagram Check Growth

Untuk memeriksa tumbuh kembang anak, proses diawali dengan *user* mengakses menu Periksa. Kemudian sistem akan menampilkan *activity* Periksa yang merupakan penampung fitur periksa. Pada halaman ini terdapat *float action button* (FAB) yang mengarahkan *user* ke halaman periksa yaitu *activity* *NewHistory*. Ketika *user* mengakses *activity* *NewHistory*, sistem akan menampilkan data nama, jenis kelamin, dan tanggal lahir dari profil anak aktif. Apabila *user* belum *login*, *user* tetap dapat melakukan pemeriksaan namun harus mengisi formulir secara lengkap. Ketika formulir sudah terisi dan tombol "Simpan" ditekan, sistem akan melakukan validasi input dan kalkulasi hasil pemeriksaan dan menyimpan data tersebut pada objek dari kelas *HistoryModel*. Kemudian, sistem akan menjalankan fungsi untuk menyimpan hasil periksa pada *database* berdasarkan profil anak aktif. Setelah data tersimpan, sistem akan menampilkan hasil pemeriksaan dengan *activity* *DetailHistory*. Bagi *user* yang tidak melakukan *login*, data akan langsung ditampilkan tanpa disimpan dalam *database*.

3.2.4 Class Diagram

Class diagram merupakan salah satu diagram dari UML yang berguna untuk menggambarkan secara jelas struktur dan hubungan setiap objek. Untuk setiap kelas dalam diagram ini memiliki komponen berupa nama, atribut, dan metode[45].



Gambar 3.18. Class Diagram

Pada class diagram diatas, terdapat 11 class activity yang terdiri dari MainActivity, Login, Register, Article, DetailArticle, Menu, MenuDetail, ChildProfile, Periksa, NewHistory, dan DetailHistory. Login memiliki hubungan asosiasi dengan Register karena halaman register hanya dapat diakses melalui halaman login.

MainActivity merupakan halaman utama pada aplikasi yang saling berasosiasi dengan *Article*, *Menu*, dan *Periksa*. Hal ini memungkinkan antar *activity* dapat saling mengakses untuk membuka tampilan masing-masing fitur. *MainActivity* juga memiliki hubungan agregasi dengan *ChildModel* yang berfungsi sebagai objek penampung data anak yang terdaftar dari *database*. Salah satu dari data tersebut dapat dijadikan profil anak aktif pada *activity* ini.

Activity Article merupakan menu untuk memilih artikel yang ingin dibaca. Daftar artikel-artikel ini diperoleh dari *database* yang kemudian disimpan sebagai objek pada *ArticleModel* yang menjadikan hubungan antar *activity* ini merupakan agregasi. *Article* juga memiliki hubungan asosiasi dengan *DetailArticle* yang berguna untuk menampilkan detil dari artikel yang telah dipilih. *Activity Menu* juga memiliki hubungan dengan *MenuModel* dan *DetailMenu* yang serupa dengan hubungan pada *Activity Article*.

Activity Periksa digunakan sebagai halaman menu utama untuk melakukan pemeriksaan tumbuh kembang dan melihat riwayat pemeriksaan. Maka dari itu, *Periksa* memiliki hubungan asosiasi dengan *DetailHistory* dan *NewHistory*. *Periksa* memiliki hubungan agregasi dengan *HistoryModel*. Pada *Periksa* juga memungkinkan adanya hubungan agregasi dengan *ChildModel* jika *user* telah *login* dan memilih profil anak aktif, sehingga *Periksa* dapat menarik data riwayat periksa dan menyimpannya pada *HistoryModel* sebagai objek penampung.

Activity NewHistory digunakan untuk menerima input untuk periksa tumbuh kembang anak dan juga melakukan kalkulasi serta menjalankan *forward chaining*. *Activity* ini memiliki hubungan agregasi dengan *ChildModel* dan *HistoryModel* yang membantu menyimpan data sebagai objek untuk proses pemeriksaan. *NewHistory* juga berasosiasi dengan *DetailHistory* untuk menampilkan hasil dari pemeriksaan. Sedangkan *DetailHistory* yang digunakan untuk menampilkan detil dari riwayat pemeriksaan atau hasil pemeriksaan memiliki hubungan agregasi dengan *HistoryModel* sebagai objek untuk membaca data dari suatu hasil pemeriksaan.

3.2.5 Database

Terdapat 2 buah basis data yang digunakan dalam pembuatan aplikasi NutriMo, diantaranya Room Database untuk menyimpan standar perhitungan antropometri untuk nilai PB/U dan BB/TB. Sedangkan, Firebase Realtime Database sebagai penyimpan informasi *user*, profil anak, artikel, menu MP-ASI dan hasil

pemeriksaan.

A Struktur Database RoomDB

Tabel 3.2 merupakan tabel data dari standar antropometri anak untuk pengukuran ambang batas atau skor-Z dari panjang badan dibandingkan usia.

Tabel 3.2. Tabel *Database* Standar Antropometri PB/U

Nama Kolom	Storage Class	Keterangan
idHaz	INTEGER	<i>Primary Key, Auto Increment</i>
gender	TEXT	Jenis kelamin anak
usia	INTEGER	Usia anak
nsd3	REAL	Nilai -3 SD
nsd2	REAL	Nilai -2 SD
nsd1	REAL	Nilai -1 SD
median	REAL	Nilai Median
psd1	REAL	Nilai +1 SD
psd2	REAL	Nilai +2 SD
psd3	REAL	Nilai +3 SD

Tabel 3.3 merupakan tabel data dari standar antropometri anak untuk pengukuran ambang batas atau skor-Z dari berat badan dibandingkan panjang badan.

Tabel 3.3. Tabel *Database* Standar Antropometri BB/PB

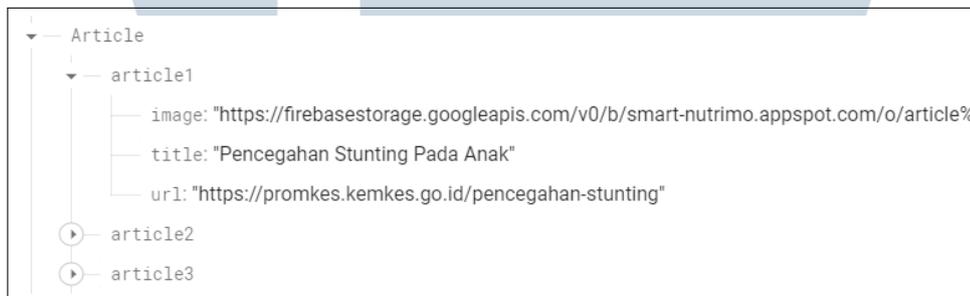
Nama Kolom	Storage Class	Keterangan
idWhz	INTEGER	<i>Primary Key, Auto Increment</i>
gender	TEXT	Jenis kelamin anak
tinggi	REAL	Panjang badan anak
nsd3	REAL	Nilai -3 SD
nsd2	REAL	Nilai -2 SD
nsd1	REAL	Nilai -1 SD
median	REAL	Nilai Median
psd1	REAL	Nilai +1 SD
psd2	REAL	Nilai +2 SD
psd3	REAL	Nilai +3 SD

B Struktur Firebase Realtime Database

Firebase Realtime Database merupakan basis data *non-SQL cloud* yang dikembangkan oleh Google. Data yang disimpan dan tersinkron secara *realtime* dalam format JSON. Dalam aplikasi NutriMo, basis data ini berfungsi sebagai pusat penyimpanan informasi pengguna dan aplikasi. Untuk data berupa gambar atau *image* akan disimpan dalam Firebase Storage yang akan direferensikan menggunakan URL.

1. Data Artikel

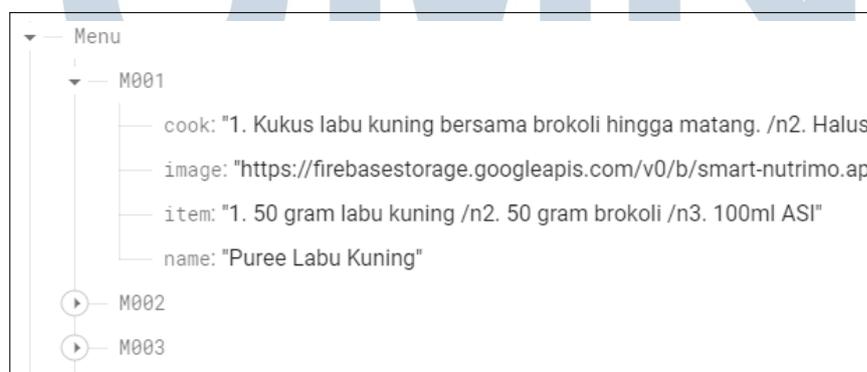
Data artikel memuat informasi terkait artikel yang disediakan pada aplikasi. Data yang disimpan berupa judul, gambar, dan URL yang dapat dilihat pada Gambar 3.19.



Gambar 3.19. Struktur Firebase Realtime Database Artikel

2. Data Menu MP-ASI

Data menu memuat informasi terkait menu yang disediakan pada aplikasi. Data yang disimpan berupa cara memasak, gambar, bahan dan nama menu yang dapat dilihat pada Gambar 3.20.



Gambar 3.20. Struktur Firebase Realtime Database Menu MP-ASI

3. Data Anak

Data anak disimpan dibawah data *User ID* yang didapatkan dengan

pembuatan akun *user* menggunakan Firebase Authentication. Data anak akan memiliki ID atau *KEY* yang dibuat secara otomatis oleh sistem. Data anak yang disimpan berupa tanggal lahir, jenis kelamin, panjang badan saat lahir, berat badan saat lahir dan nama anak. Gambar 3.21 merupakan Struktur JSON Anak.



Gambar 3.21. Struktur Firebase Realtime Database Anak

4. Data Riwayat Periksa

Seperti ID data anak, ID atau *KEY* data riwayat periksa juga dibuat otomatis oleh sistem yang disimpan di bawah ID data anak. Data riwayat menyimpan data umur anak dalam bulan, tanggal pemeriksaan, nilai ambang batas PB/U, tinggi saat periksa, kondisi tinggi anak, kondisi gizi anak, hasil rekomendasi, berat saat periksa dan nilai ambang batas BB/PB. Struktur JSON dapat dilihat pada Gambar 3.22.

U M I N
U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

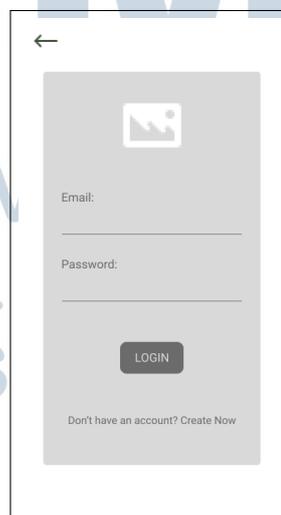


Gambar 3.22. Struktur Firebase Realtime Database Riwayat Periksa

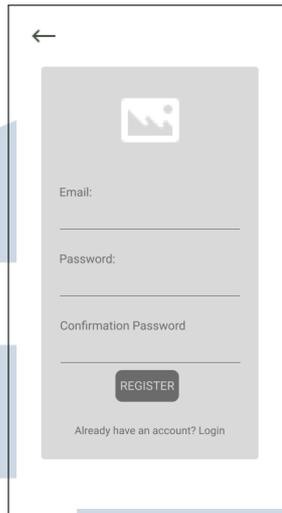
3.2.6 Rancangan Tampilan

A Autentikasi

Pada halaman autentikasi, *user* yang tidak memiliki akun dapat membuat akun baru dengan mengisi beberapa data yang diperlukan seperti pada Gambar 3.24 yaitu *email*, kata sandi, dan melakukan konfirmasi kembali kata sandi. Sedangkan apabila *user* sudah memiliki akun dapat langsung melakukan *login* dengan memasukkan alamat *email* dan kata sandi yang telah terdaftar.



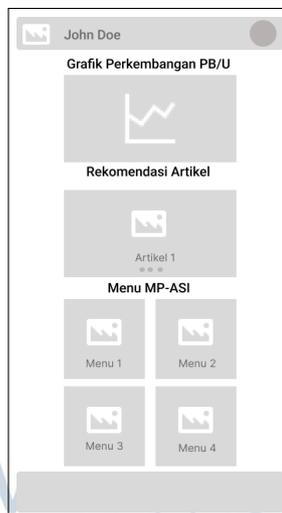
Gambar 3.23. Rancangan tampilan Login



Gambar 3.24. Rancangan tampilan Register

B Menu Utama

Untuk tampilan menu utama dapat terlihat pada Gambar 3.25 dimana akan ditampilkan grafik dari perkembangan anak, rekomendasi artikel-artikel mengenai *stunting*, ASI, dan MP-ASI, serta terdapat menu-menu MP-ASI pilihan.



Gambar 3.25. Rancangan tampilan Halaman Utama

C Profil Anak

Untuk dapat melakukan pengecekan status gizi dari pertumbuhan anak, *user* dapat menambahkan profil anak dengan mengisi data-data yang diperlukan seperti yang terlihat pada Gambar 3.26.

Gambar 3.26. Rancangan tampilan Tambah Profil Anak

Gambar 3.27. Rancangan tampilan Edit Profil Anak

D Menu Periksa

Setelah memasukkan profil anak, terdapat menu periksa untuk memantau status perkembangan anak yang akan divisualisasikan dengan grafik seperti yang terlihat pada Gambar 3.28. Untuk melakukan pemeriksaan, *user* melakukan input data terbaru dengan mengisi beberapa data mengenai pertumbuhan anak yang kemudian akan diolah oleh sistem untuk mengetahui hasil dari pertumbuhan anak.



Gambar 3.28. Rancangan tampilan Periksa

A mobile application form for adding a new growth examination record. It features a back arrow at the top left. The form fields are: 'Periksa Tumbuh Kembang Anak' (Child Growth Examination), 'Nama Lengkap' (Full Name) with the value 'John Doe', 'Jenis Kelamin' (Gender) with the value 'Laki-laki', 'Tanggal Periksa' (Examination Date) with the value '4/4/2022', 'Berat Badan saat ini (Kg)' (Current Weight in Kg) with the value '3.2', and 'Tinggi Badan saat ini (Cm)' (Current Height in Cm) with the value '45'. A 'SIMPAN' (Save) button is located at the bottom.

Gambar 3.29. Rancangan tampilan Tambah Riwayat Periksa

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.30. Rancangan tampilan Riwayat Periksa

E Menu Artikel

Pada menu artikel, akan ditampilkan headline dari beberapa artikel terpilih mengenai *stunting*, ASI, dan MP-ASI yang bertujuan untuk menambah pengetahuan para *user*. Apabila *user* memilih untuk membuka salah satu artikel maka akan diarahkan kepada tampilan *webview* seperti pada Gambar 3.32.



Gambar 3.31. Rancangan tampilan Menu Artikel



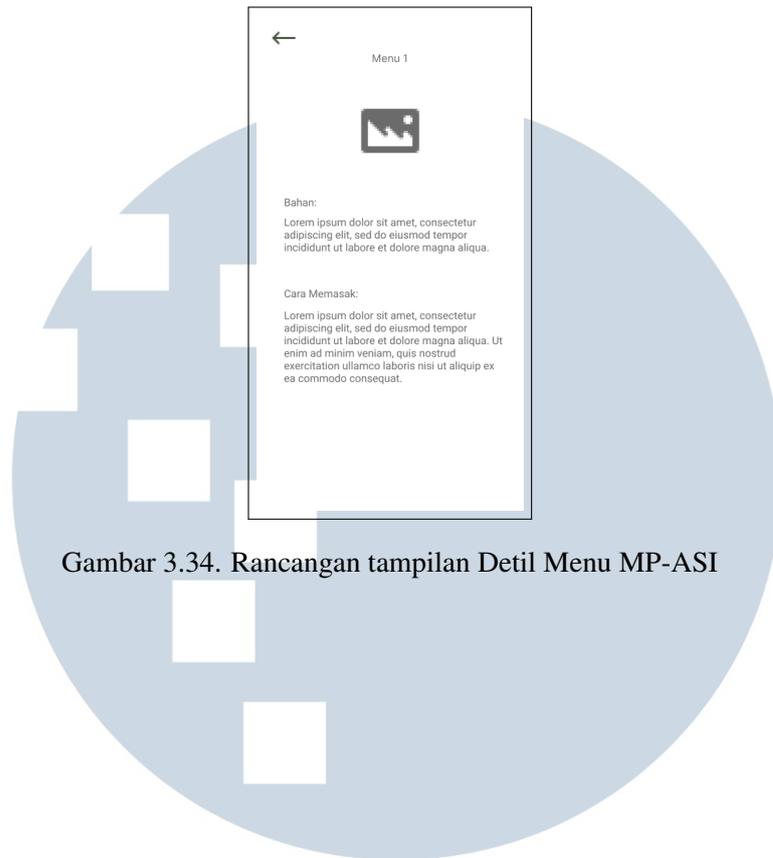
Gambar 3.32. Rancangan tampilan Detil Artikel

F Menu MP-ASI

Untuk menu MP-ASI, diberikan beberapa pilihan menu masakan yang dapat menjadi opsi untuk para *user* dalam meningkatkan gizi dari pertumbuhan anak. Dalam menu MP-ASI ini, disediakan informasi berupa bahan dan cara memasak makanan yang akan terlihat seperti Gambar 3.34.



Gambar 3.33. Rancangan tampilan Menu MP-ASI



Gambar 3.34. Rancangan tampilan Detil Menu MP-ASI

UMMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA