BAB 3
ANALISIS DAN PERANCANGAN

3.1 Analisis Sistem

Analisis sistem dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh ke dalam bagian – bagian komponennya dengan maksud untuk mengidentifikasi dan mengevaluasi permasalahan – permasalahan, kesempatan – kesempatan, hambatan – hambatan yang terjadi dan kebutuhan – kebutuhan yang diharapkan sehingga dapat sesuai dengan kebutuhan.

3.1.1 Analisis Masalah

Analisis masalah merupakan langkah pertama yang harus dilakukan dalam tahap analisis sistem. Masalah dapat didefinisikan sebagai suatu pertanyaan yang diinginkan untuk dipecahkan. Oleh karena itu langkah pertama yang harus dilakukan pada tahap ini adalah mengidentifikasi terlebih dahulu masalah-masalah yang terjadi.

Berdasarkan hasil pengolahan data kuesioner yang telah diberikan kepada 30 responden yang berusia 12 sampai 25 tahun tentang melatih daya ingat, terdapat permasalahan mengenai terbatasnya lembaga yang menyediakan fasilitas untuk melatih kemampuan daya ingat dan kurangnya pengetahuan cara melatih daya ingat. Adapun permasalahan yang terjadi yaitu sebagai berikut:

1. Terbatasnya lembaga yang menyediakan fasilitas untuk melatih kemampuan daya ingat. Dengan terbatasnya fasilitas ini, maka masyarakat akan kesulitan dalam melatih kemampuan daya ingat sehingga dibutuhkan adanya tambahan fasilitas, salah satunya berupa aplikasi untuk melatih kemampuan daya ingat yang berbasis *mobile*, agar pengguna dapat melakukan latihan daya ingat di manapun dan kapanpun.
2. Kurangnya pengetahuan tentang cara melatih kemampuan daya ingat merupakan salah satu faktor yang dapat menyebabkan menurunnya kemampuan daya ingat. Oleh karena itu, dibutuhkan penjelasan mengenai bagaimana cara atau metode untuk melatih kemampuan daya ingat.

3.1.2 Analisis Prosedur yang Berjalan

Sebelum merancang aplikasi yang akan dibuat, terlebih dahulu dilakukan analisis pada prosedur yang sedang berjalan. Hal ini bertujuan agar aplikasi yang akan dibangun nanti sesuai dengan cakupan sistem yang sudah ada.

Metode latihan daya ingat yang akan digunakan adalah dengan metode n-back. Metode ini telah digunakan dalam berbagai penelitian ilimiah untuk meningkatkan kemampuan daya ingat. Pada penerapannya saat ini, latihan dengan metode n-back ini menggunakan sistem yang telah terkomputerisasi dan diterapkan di *desktop*. Dalam prosedurnya, peserta latihan diberikan jadwal latihan dalam beberapa hari. Kemudian pada saat latihan, peserta latihan ditampilkan stimulus visual atau rangkaian *item* dan peserta latihan diminta untuk memilih *item* yang sesuai dengan *item* sebelumnya sesuai dengan posisi n sebelumnya.

Adapun aplikasi latihan daya ingat dengan metode n-back di sistem Android yang sudah ada, seperti “Single N-Back” yang merupakan aplikasi untuk melatih daya ingat dengan metode *single* n-back. Aplikasi ini dipilih untuk dianalisis karena menggunakan metode yang sama seperti aplikasi yang akan dibangun. Berdasarkan hasil pengolahan data kuesioner yang telah diberikan kepada 30 responden yang berusia 12 sampai 25 tahun mengenai penggunaan aplikasi “Single N-Back”, 80% responden merasa bingung ketika menggunakan aplikasi dan 77% responden membutuhkan penjelasan latihan. Dari data tersebut dapat ditarik kesimpulan bahwa mayoritas pengguna merasa bingung ketika menggunakan aplikasi tersebut, salah satu alasannya adalah dikarenakan pada aplikasi tersebut tidak ada penjelasan mengenai cara melakukan latihan daya ingat dengan metode n-back.

3.1.3 Analisis Metode

Analisis metode menggambarkan metode yang digunakan dalam latihan dan pengujian kemampuan daya ingat yang akan diterapkan pada aplikasi. Berikut

ini adalah analisis metode yang terdiri dari analisis metode n-back dan analisis pengujian kemampuan daya ingat:

3.1.3.1 Analisis Metode N-Back

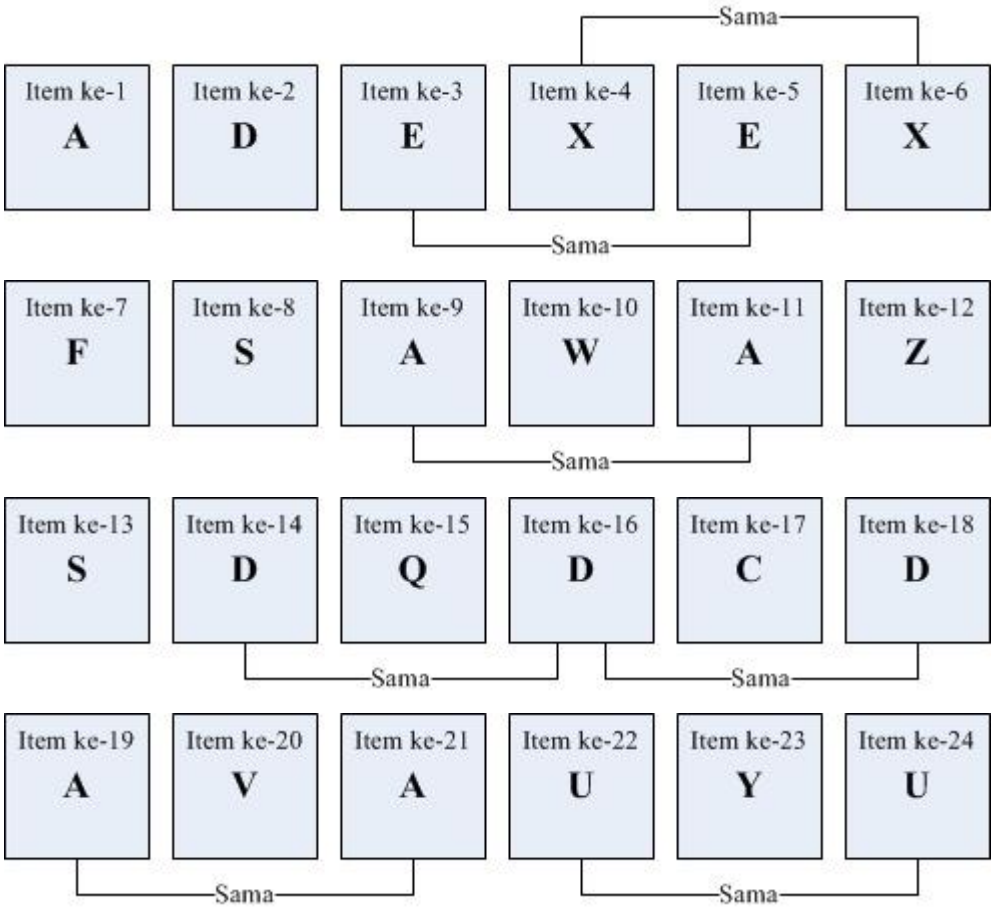
Latihan yang akan diterapkan menggunakan metode n-back. Metode ini umum digunakan untuk melatih memori kerja. Metode n-back menggunakan beberapa *item* yang akan ditampilkan secara bergantian. Peserta latihan diminta untuk memilih *item* mana yang sesuai dengan *item* sebelumnya sesuai dengan n yang digunakan. Metode ini cocok untuk melatih daya ingat, berdasarkan tiga penelitian ilmiah sebelumnya yang telah meneliti mengenai latihan daya ingat dengan metode n-back ini, berikut kesimpulan dari tiga penelitian tersebut:

1. Terdapat peningkatan pada memori kerja dan *fluid intelligence* dilihat dari skor tes sebelum dan sesudah latihan. *Fluid intelligence* adalah kemampuan seseorang untuk berpikir secara logika [7].
2. Terdapat lima efek peningkatan yang berkaitan dengan memori kerja dari latihan dengan metode n-back, yaitu peningkatan pada perhatian khusus, peningkatan dalam memperbaharui informasi, peningkatan pada pergantian fokus, peningkatan pada fokus perhatian, dan peningkatan kapasitas memori kerja [8].
3. Terdapat peningkatan terhadap skor tes memori kerja sebelum dan sesudah latihan. Tetapi, peningkatan tersebut didapat dari kelompok yang melakukan latihan n-back dengan terjadwal [9].

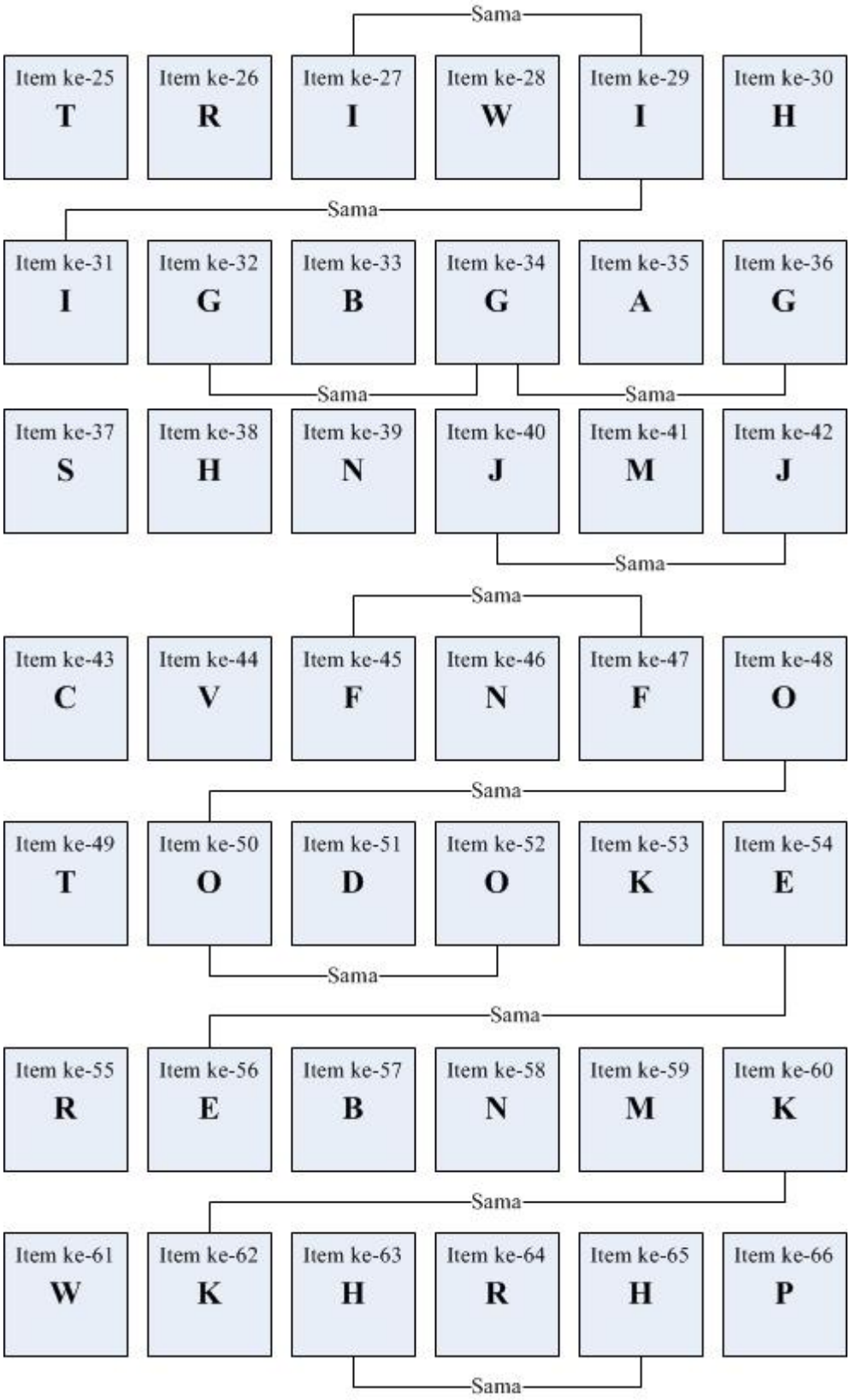
Untuk nilai n yang digunakan pada metode n-back yang akan diterapkan yaitu 2 sampai 4. *Item* yang digunakan yaitu, dapat berupa angka, huruf, dan juga kata. Jumlah *item* untuk tiap latihan dihitung dengan rumus: (20 + n) x 3. *Item* ditampilkan secara bergantian dengan *delay* antar *item* 2,5 detik pada latihan n-back yang telah terkomputerisasi [7].

Berikut ini adalah proses-proses yang terjadi pada saat melakukan latihan daya ingat dengan metode n-back, kasus yang digunakan yaitu menggunakan nilai n = 2 dan konten yang digunakan yaitu huruf:

1. Langkah pertama, menghitung jumlah *item* yang akan digunakan. Dihitung dengan rumus (20 + n) x 3, n = 2, maka (20 + 2) x 3 = 66.
2. Langkah kedua, menentukan posisi *item* yang sama. Jumlah *item* yang sama yaitu 18 *item*. Untuk posisi *item* yang sama diacak, sehingga untuk tiap latihan menggunakan posisi yang sama yang berbeda. Contohnya didapat posisi *item* yang sama yaitu, posisi ke - 3, 4, 9, 14, 16, 19, 22, 27, 29, 32, 34, 40, 45, 48, 50, 54, 60, 63.
3. Langkah ketiga, menentukan *item – item* yang akan digunakan. Karena pada contoh kasus ini menggunakan huruf maka *item* yang digunakan dari A sampai Z. Acak huruf yang akan ditampilkan, sehingga didapat susunan *item* yang dapat dilihat pada Gambar 3.1 dan Gambar 3.2



Gambar 3.1 Susunan Item ke-1 Sampai Item ke-24



Gambar 3.2 Susunan Item ke-25 Sampai Item ke-66

3.1.3.2 Analisis Pengujian Kemampuan Daya Ingat

Pengujian kemampuan daya ingat yang akan digunakan adalah dilihat dari kemampuan memori kerja seseorang dalam menyelesaikan tes yang disajikan. Tes yang akan digunakan adalah dengan menggunakan *Reading Span Test* (RST). Tes ini dirancang sesuai dengan karakteristik memori kerja, dimana karakteristik dari memori kerja, yaitu mengolah informasi yang diterima, apakah harus diingat atau dilupakan.

Reading Span Test menggunakan 60 kalimat yang tidak saling berkatian dan ditampilkan secara bergantian. Tes ini dibagi menjadi 5 bagian. Pada bagian pertama menggunakan 6 kalimat, pada bagian kedua menggunakan 9 kalimat, pada bagian ketiga menggunakan 12 kalimat, pada bagian keempat menggunakan 15 kalimat, pada bagian kelima menggunakan 18 kalimat. Pada tiap bagian dibagi menjadi 3 set. Peserta diminta untuk membaca keseluruhan teks yang ditampilkan dan mengingat kata terakhir dari tiap kalimat. Pada akhir set, peserta diminta untuk mengeluarkan kembali kata terakhir pada tiap kalimat yang disajikan.

Reading Span Test telah teruji validitasnya dalam mengukur kemampuan memori kerja. Tes ini merupakan salah satu tes yang paling banyak digunakan di dunia dalam mengukur kemampuan memori kerja. Adapun skoring yang digunakan pada tes ini yaitu dengan menghitung persentase dari tiap bagian. Skoring dengan menghitung persentase untuk tiap bagian merupakan skoring yang paling baik [19]. Dengan sistem skoring seperti ini, maka melupakan kata-kata pada bagian awal akan menghasilkan nilai yang lebih rendah dibandingkan dengan pada bagian akhir, misalnya jika peserta tes melupakan 3 kata pada bagian pertama maka hasil yang diperoleh adalah 50%, sedangkan jika peserta tes melupakan 3 kata pada bagian kelima maka hasil yang diperoleh adalah 83%. Adapun proses-proses yang terjadi pada *Reading Span Test* (RST), yaitu sebagai berikut:

1. Langkah pertama, menentukan susunan 60 kalimat yang akan digunakan secara acak. Kalimat tersebut dibagi menjadi 5 bagian, pada tiap bagian dibagi menjadi 3 set. Contoh susunan kalimat dapat dilihat pada Tabel 3.1.

Tabel 3.1 Susunan Kalimat

Bagian	Set	Kalimat
1	1	Setelah itu, selamatlah ia dari kejaran massa dan polisi negara yang akan menangkapnya.
		Sangkuriang merupakan sebuah legenda yang sangat terkenal di kalangan masyarakat Sunda di Jawa Barat, Indonesia.
	2	Dalam beberapa hari terakhir, Prabu Tapa Agung terlihat sering duduk termenung seorang diri di atas singgasananya.
		Masha berjalan mencari jalan pulang sampai akhirnya dia menemukan pondok kecil yang dihuni oleh beruang.
	3	Tak terasa air mata nenek itu bercucuran membasahi kedua pipinya yang sudah keriput.
		Akhirnya pohon oak itu berbaring di antara alang-alang yang menatapnya dengan iba.
2	1	Tak heran jika penduduk sekitarnya banyak yang jatuh miskin karena tanah pertanian mereka habis.
		Prabu Siliwingi heran melihat penyakit itu tiba-tiba menyerang putri dan permaisurinya secara bersamaan.
		Sementara itu, Nini Arga yang melihat keadaan suaminya seperti itu berusaha menasehati agar mengurungkan niatnya.
	2	Ketika berada dalam jarak yang cukup, burung hantu tua segera menerkam belalang.
		Ia benar-benar tidak pernah mengira sebelumnya jika ia akan mengalami nasib seperti itu.
		Akhirnya, kuda-kuda itu sama sekali tidak bisa bergerak saat salah satu roda tenggelam dalam lumpur.

Bagian	Set	Kalimat
	3	Karena itu, teman ayahnya meminta salah seorang dari mereka untuk mengambilkan sapu lidi.
		Dua orang pelancong, yang sedang kegerahan, mencari tempat berteduh dan beristirahat di bawah pohon.
		Anak laki-laki itu menggenggam banyak sekali kacang, sehingga ketika hendak dikeluarkan tangannya tersangkut.
3	1	Seekor elang itu kemudian mencengkeram domba dengan cakarnya dan membawa kabur ke sarangnya.
		Setelah sempat bingung, akhirnya Aki Panyumpit memutuskan untuk berburu binatang ke hutan sebelah timur.
		Rara Anteng akan mengajukan satu persyaratan yang kira-kira tidak sanggup dipenuhi oleh raksasa.
		Dengan kesepatan bersama tiga binatang lainnya, singa membagi hasil buruannya menjadi empat bagian yang sama besar.
	2	Tidak disangka dia justru menemukan topeng di antara tumpukan-tumpukan barang yang dimiliki aktor.
		Seekor rubah melihat seekor gagak berdiri di atas ranting dengan sepotong keju di paruhnya.
		Mereka kerap bertarung hingga berhari-hari lamanya, namun tidak pernah ada yang kalah maupun menang.
		Tepuk tangan penonton pun kembali bergema, sedangkan Putri Dyah Ayu Purpasari hanya terdiam.
	3	Menurut legenda yang beredar di kalangan masyarakat setempat, ketinggian gunung ini dahulu hampir mencapai langit.
		Seekor anjing sedang berjalan pulang sambil membawa makan malam, sebuah daging besar, di mulutnya.
		Ketika datang, tikus kota kecewa dengan makanan yang

Bagian	Set	Kalimat
		tidak lebih dari beberapa biji jagung.
		Ratu Ayu Kencana Wungu sangat khawatir ketika mendengar bahwa Minakjingga ingin menyerang kerajaannya.
4	1	Saat Raja Serule masuk ke dalam istana tersebut, Sangede menunggu di halaman istana.
		Setelah beberapa saat melihat dunia, dia kembali untuk memberitahu ibunya dua makhluk indah yang telah dilihatnya.
		Ketika sudah tiba waktunya bagi mereka untuk bekerja setiap hari, sehingga mereka bisa mencari nafkah.
		Pada sore hari, anak-anak mulai melemparkan batu pada katak, bahkan saling berkompetisi.
		Atas kejadian tersebut, maka suatu hari suaminya berpura-pura mati sehingga anaknya menangis.
	2	Pada tahun sebelum masehi, Aesop adalah seorang pendongeng cerita anak-anak yang terkenal dengan cerita fabel.
		Warga desa pun menghentikan aktivitas mereka dan segera menyelamatkan domba-domba rahang serigala ganas.
		Ketika beruang mengintai sarang itu, seekor lebah kecil terbang keluar dari sarang untuk melindungi kawanannya.
		Alkisah, di Negeri Nanggroe Aceh Darussalam, Indonesia, hiduplah seorang Raja yang adil dan bijaksana.
		Setibanya di gubuk, ia langsung menghampiri anak semata wayangnya yang sedang terbaring lemas.
	3	Konon, di tengah hutan belantara itu, hiduplah sekawanan burung parakeet yang hidup damai, tenteram, dan makmur.
		Berikutnya pun sama, waktu menyalakan korek api

Bagian	Set	Kalimat
		berikutnya, dia melihat pohon natal yang sangat indah.
		Keistimewaan lain yang dimiliki oleh Kabupaten Simeulue adalah keanekaragaman budayanya, seperti bahasa, upacara-upacara tradisional.
		Ketika melewati pepohonan di dekat pabrik, ia melihat seorang pemuda duduk bermalas-malasan di bawahnya.
		Dari kejauhan, tampaklah seorang laki-laki setengah baya yang sedang menyiangi rumput di tengah-tengah ladangnya.
5	1	Ia kemudian duduk di bawah pohon dekat telaga melepas lelah, angin sepoi-sepoi membuatnya tertidur.
		Tapi ia begitu terkejut, ketika melihat kudanya tak ada di tempat yang seharusnya.
		Pangeran Mangkubumi adalah pahlawan nasional yang merupakan ikon yang tak bisa dilepaskan dari masyarakat.
		Setelah dua hari duduk di atas keledainya tanpa arah tujuan dengan bekal menipis.
		Di negeri itu terjadi musibah kelaparan besar, makanan yang tersisa di rumah hanya sedikit sekali.
		Ada sebuah cerita yang mengisahkan tentang saudagar kaya, yang mempunyai sifat kikir dan sombong.
	2	Suatu hari, Kerajaan yang berada di bawah pimimpin Pamuncak Rencong Talang berlimpah hasil panennya.
		Ketika Aminah mendekati bibir sungai, buaya perompak segera menyambarnya kemudian dengan cepat membawanya pergi.
		Keduanya masih dipercaya menjaga Beringin Putih di utara masjid yang ada di selatan jalan.
		Suatu ketika, Putri diajak Sari ibunya berbelanja di pasar yang terletak lumayan jauh dari rumahnya.

Bagian	Set	Kalimat
		Tawaran yang diajukan tersebut harus diterima, jika tidak ia akan membuat dusunnya binasa.
		Dan tepat ketika itu, mereka semua melihat Santana mengangkat bangkai Oarfish ke darat.
	3	Ladang yang tumbuh subur pun perlahan-lahan mulai mengering dan hasilnya tidak seperti biasanya.
		Dalam cerita rakyat yang berasal dari Kalimantan Tengah dikisahkan ada seorang raja di pedalaman.
		Dia putra Raja Hindu yang datang ke Pamona dan membuat ramai kawasan setempat.
		Rupanya itu suara seorang perempuan yang sedang dicengkeram oleh elang yang sedang terbang.
		Di tengah perjalanan, kapal yang mereka tumpangi oleng akibat diterjang ombak yang begitu besar.
		Tak lama kemudian Ariel bertemu dengan Pangeran Erik, sejak mendengarnya bernyanyi di suatu malam.

2. Langkah kedua, pada akhir set peserta pengujian akan menjawab dua kata terakhir pada tiap kalimat.
3. Langkah ketiga, skoring dilakukan pada akhir tiap bagian dengan menggunakan persentase dari kata yang dapat diingat pada bagian tersebut. Perhitungan persentase kebenaran dari kata yang dapat diingat dapat dilihat pada Tabel 3.2.

Tabel 3.2 Skoring Tes

Bagian	Jumlah Kata yang Benar	Skor
1	6	100%
	5	83%
	4	67%
	3	50%

Bagian	Jumlah Kata yang Benar	Skor
	2	33%
	1	17%
2	9	100%
	8	89%
	7	78%
	6	67%
	5	56%
	4	44%
	3	33%
	2	22%
	1	11%
3	12	100%
	11	92%
	10	83%
	9	75%
	8	67%
	7	58%
	6	50%
	5	42%
	4	33%
	3	25%
	2	17%
	1	8%
4	15	100%
	14	93%
	13	87%
	12	80%
	11	73%
	10	67%

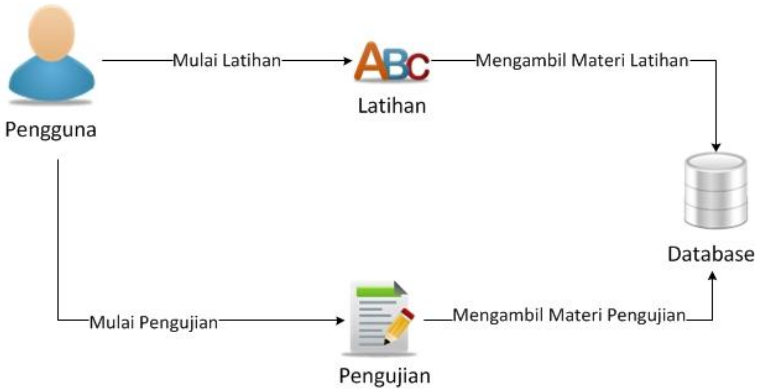
Bagian	Jumlah Kata yang Benar	Skor
	9	60%
	8	53%
	7	47%
	6	40%
	5	33%
	4	27%
	3	20%
	2	13%
	1	7%
5	18	100%
	17	94%
	16	89%
	15	83%
	14	78%
	13	72%
	12	67%
	11	61%
	10	56%
	9	50%
	8	44%
	7	39%
	6	33%
	5	28%
	4	22%
	3	17%
	2	11%
	1	6%

3.1.4 Analisis Kebutuhan Data

Analisis kebutuhan data menggambarkan data apa saja yang akan diterapkan pada aplikasi. Adapun data yang digunakan, yaitu data materi, data jadwal latihan, dan data pengujian. Penyimpanan dan pengambilan data-data tersebut menggunakan *database* yang terdapat pada aplikasi. Berikut analisis data materi, data jadwal latihan dan data pengujian:

3.1.4.1 Analisis Data Materi

Data materi meupakan materi-materi yang akan digunakan dalam latihan maupun pengujian kemampuan daya ingat. Pada data materi terdapat tiga atribut, yaitu id materi, nama materi, dan isi materi. Nama materi yang digunakan ada empat, yaitu angka, huruf, kata, dan kalimat. Isi materi menggunakan *database* yang di dalamnya terdapat isi ataupun konten dari materi, misalnya untuk huruf isinya yaitu huruf A sampai Z. Data materi ini disimpan di dalam *database* yang terdapat pada aplikasi dan telah terinisialisasi sebelumnya. Sistem nantinya hanya membaca data materi tersebut. Adapun alur dalam menampilkan data materi dapat dilihat pada Gambar 3.3.



Gambar 3.3 Alur Data Materi

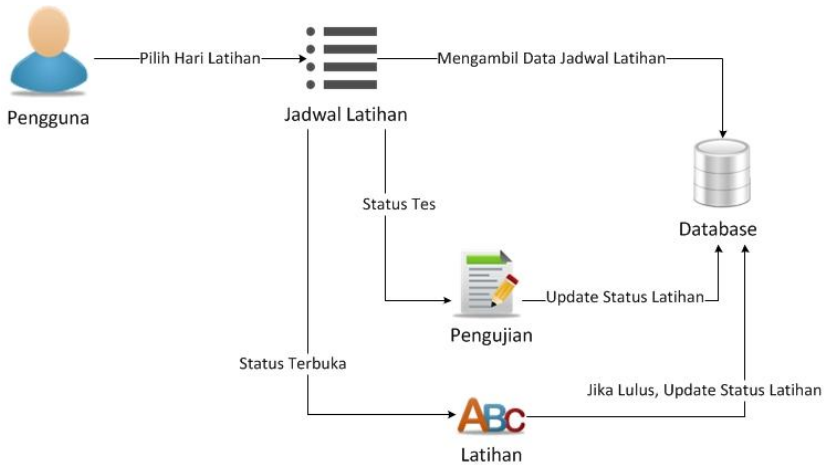
Pada saat pengguna memulai latihan maka materi latihan yang diambil yaitu dapat berupa angka, huruf, dan latihan. Sedangkan, pada saat pengguna memulai pengujian maka materi pengujian yang diambil yaitu berupa kalimat.

3.1.4.2 Analisis Data Jadwal Latihan

Data jadwal latihan merupakan daftar hari latihan yang nantinya akan diterapkan pada sistem. Pada data jadwal latihan terdapat tiga atribut, yaitu hari latihan, id materi, dan juga status latihan. Data jadwal latihan ini berelasi dengan data materi. Data pada jadwal latihan terdiri dari 30 data, dikarenakan latihan yang akan diterapkan adalah selama 30 hari. Pada jadwal latihan terdapat empat status latihan yang digunakan, diantaranya yaitu:

1. "Tes", maksudnya yaitu pengguna harus melakukan tes kemampuan daya ingat terlebih dahulu.
2. "Terbuka", maksudnya yaitu pengguna telah dapat melakukan latihan pada hari tersebut.
3. "Lulus", maksudnya yaitu pengguna telah lulus pada hari latihan tersebut.
4. "Terkunci", maksudnya yaitu pengguna belum dapat melakukan latihan pada hari tersebut dikarenakan ada latihan yang belum lulus pada hari sebelumnya.

Adapun alur dalam menampilkan dan memperbaharui data jadwal latihan, yaitu sebagai berikut:



Gambar 3.4 Alur Data Jadwal Latihan

Sistem akan mengambil data jadwal latihan dari *database* dan kemudian menampilkannya kepada pengguna. Pada saat pengguna memilih hari latihan, jika status latihan yang dipilih "Terbuka" maka pengguna akan melakukan latihan jika lulus latihan tersebut, maka sistem akan *update* data status latihan menjadi

"Lulus" dan *update* data status latihan setelahnya menjadi "Terbuka". Sedangkan, jika status latihan yang dipilih "Tes" maka pengguna akan melakukan tes, setelah menyelesaikan tes tersebut, sistem akan *update* status latihan yang dipilih menjadi "Terbuka".

3.1.4.3 Analisis Data Pengujian

Data pengujian merupakan daftar pengujian yang akan diterapkan pada sistem. Pada data pengujian terdapat dua atribut, yaitu id pengujian dan status pengujian. Status pengujian yang digunakan yaitu "Sudah Tes" dan "Belum Tes". Data pengujian ini berelasi dengan data hasil pengujian yang berisi hasil pengujian dari tiap bagian dan juga berelasi dengan data materi. Pada hasil pengujian terdapat empat atribut yaitu id hasil, bagian, skor, id pengujian. Data pengujian ini disimpan di dalam *database* yang terdapat pada aplikasi dan telah terinisialisasi sebelumnya. Sistem nantinya membaca dan memperbaharui data pengujian tersebut dan juga memasukkan hasil pengujian. Adapun alur dalam membaca dan memperbaharui data pengujian, yaitu sebagai berikut:



Gambar 3.5 Alur Data Pengujian

Setelah pengguna menyelesaikan tes atau pengujian, kemudian sistem akan memasukkan hasil ujian ke dalam *database* dan juga *update* status pengujian menjadi "Sudah Tes". Ketika pengguna akan melihat hasil ujian, maka sistem akan mengambil data hasil ujian dari *database*.

3.1.5 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional merupakan analisis yang dibutuhkan untuk menentukan spesifikasi kebutuhan sistem. Spesifikasi ini juga meliputi elemen atau komponen-komponen apa saja yang dibutuhkan untuk sistem yang akan dibangun sampai sistem tersebut dapat diimplementasikan.

Analisis kebutuhan non fungsional menggambarkan kebutuhan sistem yang menitikberatkan pada properti perilaku yang dimiliki oleh sistem, diantaranya kebutuhan pengguna, perangkat keras, serta perangkat lunak sebagai bahan analisis kekurangan dan kebutuhan yang harus dipenuhi dalam perancangan sistem yang akan diterapkan.

3.1.5.1 Analisis Pengguna

Analisis pengguna merupakan gambaran tentang karakteristik pengguna untuk aplikasi yang akan dibangun. Karakteristik pengguna yang dianalisis berdasarkan empat kategori, yaitu pengguna, usia, hak akses, dan tingkat keterampilan. Karakteristik pengguna yang terlibat dalam aplikasi *mobile* "Latihan Daya Ingat" dapat dilihat pada Tabel 3.3.

Tabel 3.3 Karakteristik Pengguna

Pengguna	Usia	Hak Akses	Tingkat Keterampilan
Pengguna	12 – 25 tahun	Mengakses semua konten yang terdapat pada aplikasi.	Dapat mengoperasikan *smartphone* yang berbasis Android.

3.1.5.2 Analisis Perangkat Keras

Analisis perangkat keras merupakan gambaran tentang spesifikasi perangkat keras untuk pengguna dalam menjalankan aplikasi *mobile* "Latihan Daya Ingat". Adapun spesifikasi perangkat keras yang direkomendasikan untuk dapat menggunakan aplikasi *mobile* "Latihan Daya Ingat" dapat dilihat pada Tabel 3.4.

Tabel 3.4 Spesifikasi Perangkat Keras untuk Pengguna

No.	Perangkat Keras	Spesifikasi
1	CPU	800 MHz
2	*Space* hard disk	5 MB
3	RAM	512 MB
4	Dimensi Layar	4,0 inch

3.1.5.3 Analisis Perangkat Lunak

Analisis perangkat lunak merupakan gambaran tentang spesifikasi perangkat lunak untuk pengguna dalam menjalankan aplikasi *mobile* "Latihan Daya Ingat". Adapun spesifikasi perangkat lunak minimum yang dibutuhkan untuk dapat menggunakan aplikasi melatih daya ingat dengan metode n-back dan reading span test berbasis Android yaitu sistem operasi Android versi 2.3.3 (*Gingerbread*).

3.1.6 Analisis Kebutuhan Fungsional

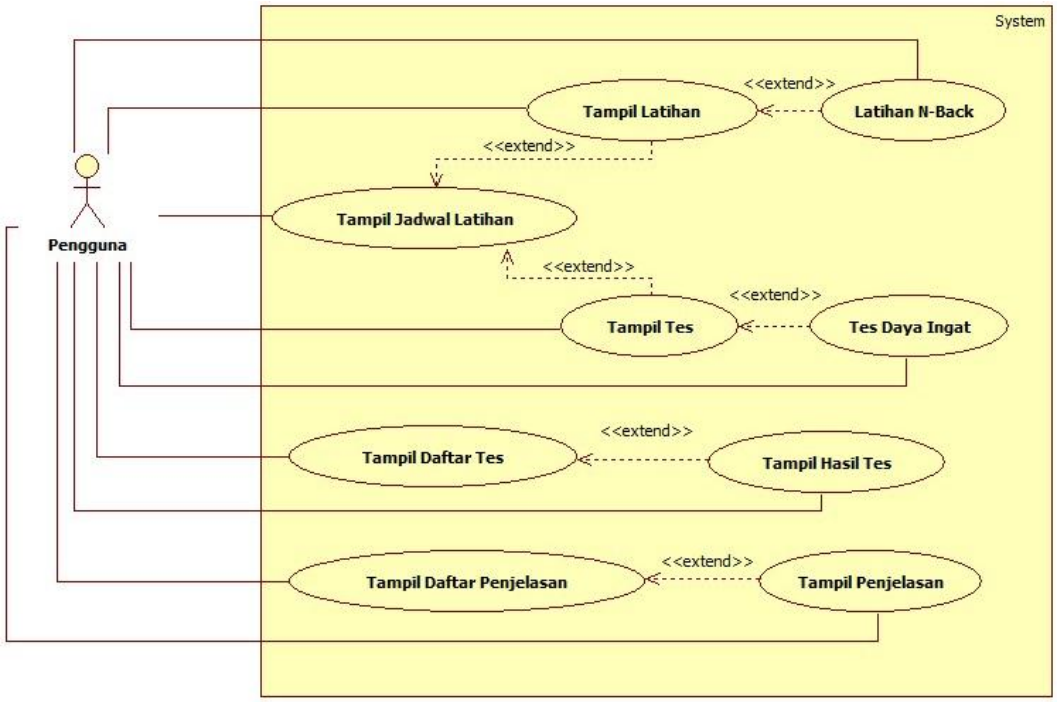
Analisis kebutuhan fungsional menggambarkan proses kegiatan yang akan diterapkan dalam sebuah sistem dan menjelaskan kebutuhan yang diperlukan sistem agar sistem dapat berjalan dengan baik serta sesuai dengan kebutuhan. Pemodelan sistem dimodelkan dengan menggunakan UML (*Unified Modeling Language*).

Tahap-tahap pemodelan sistem dalam analisis tersebut antara lain *Use Case Diagram, Activity Diagram, Sequence Diagram, Class Diagram, Component Diagram* dan *Deployment Diagram*.

3.1.6.1 *Use Case Diagram*

Use case diagram merupakan bagian tertinggi dari fungsionalitas yang dimiliki sistem yang akan menggambarkan bagaimana seseorang atau aktor akan menggunakan dan memanfaatkan sistem. Diagram ini juga mendeskripsikan apa yang akan dilakukan oleh sistem.

Use case diagram terdiri dari tiga bagian yaitu definisi aktor, identifikasi *use case* dan skenario *use case*. *Use case diagram* aplikasi "Latihan Daya Ingat" dapat dilihat pada Gambar 3.6.



Gambar 3.6 *Use Case Diagram* Aplikasi "Latihan Daya Ingat"

3.1.6.1.1 Definisi Aktor

Dalam sistem yang akan dibangun terdapat satu aktor, yaitu pengguna. Penjelasan mengenai definisi aktor dapat dilihat pada Tabel 3.5.

Tabel 3.5 Definisi Aktor

Aktor	Definisi
Pengguna	Merupakan aktor yang menggunakan aplikasi "Latihan Daya Ingat" dengan metode n-back.

3.1.6.1.2 Definisi *Use Case*

Terdapat 9 *use case* dalam aplikasi ini. Setiap *use case* menggambarkan operasi-operasi yang berbeda. Definisi *use case* yang terdapat dalam sistem dapat dilihat pada Tabel 3.6 berikut:

Tabel 3.6 Definisi *Use Case*

No.	*Use Case*	Deskripsi
1	Tampil Jadwal Latihan	Fungsionalitas untuk menampilkan jadwal latihan.
2	Tampil Latihan	Fungsionalitas untuk menampilkan latihan n-back.
3	Latihan N-Back	Fungsionalitas untuk melakukan latihan n-back.
4	Tampil Tes	Fungsionalitas untuk menampilkan tes daya ingat.
5	Tes Daya Ingat	Fungsionalitas untuk melakukan tes daya ingat.
6	Tampil Daftar Tes	Fungsionalitas untuk menampilkan daftar tes daya ingat.
7	Tampil Hasil Tes	Fungsionalitas untuk menampilkan hasil tes daya ingat.
8	Tampil Daftar Penjelasan	Fungsionalitas untuk menampilkan daftar penjelasan latihan.
9	Tampil Penjelasan	Fungsionalitas untuk menampilkan penjelasan latihan.

3.1.6.1.3 Skenario *Use Case*

Skenario *use case* mendeskripsikan urutan langkah-langkah dalam proses bisnis, baik yang dilakukan aktor terhadap sistem maupun yang dilakukan oleh sistem terhadap aktor. Berdasarkan *use case diagram* pada Gambar 3.6 sebelumnya maka skenario *use case* untuk aplikasi yang dibangun akan dijelaskan sebagai berikut:

1. Skenario *Use Case* Tampil Jadwal Latihan

Skenario *Use Case* Tampil Jadwal Latihan akan dijelaskan pada Tabel 3.7 berikut ini:

Tabel 3.7 Skenario *Use Case* Tampil Jadwal Latihan

Use case name	**Tampil Jadwal Latihan**	
Related Requirements	Tampil Latihan	
Goal In Context	Pengguna ingin menampilkan jadwal latihan selama 30 hari.	
Preconditions	Sistem menampilkan menu utama.	
Successful End Condition	Sistem berhasil menampilkan jadwal latihan selama 30 hari.	
Failed End Condition	Sistem gagal menampilkan jadwal latihan.	
Primary Actors	Pengguna	
Secondary Actors	-	
Trigger	Pengguna memilih menu “Mulai Latihan”.	
Main Flow	*Step*	*Action*
	1	Sistem menampilkan menu utama.
	2	Pengguna memilih menu “Mulai Latihan”.
	3	Sistem berhasil menampilkan jadwal latihan selama 30 hari.
Extensions	*Step*	*Branching Action*
	3.1	Sistem gagal menampilkan jadwal latihan.

2. Skenario *Use Case* Tampil Latihan

Skenario *Use Case* Tampil Latihan akan dijelaskan pada Tabel 3.8 berikut ini:

Tabel 3.8 Skenario *Use Case* Tampil Latihan

Use case name	**Tampil Latihan**	
Related Requirements	Tampil Jadwal Latihan, Latihan N-Back	
Goal In Context	Pengguna ingin menampilkan latihan n-back.	
Preconditions	Sistem menampilkan jadwal latihan.	
Successful End Condition	Sistem berhasil menampilkan latihan n-back.	
Failed End Condition	Sistem gagal menampilkan latihan n-back.	
Primary Actors	Pengguna	
Secondary Actors	-	
Trigger	Pengguna memilih hari latihan.	
Main Flow	*Step*	*Action*
	1	Sistem menampilkan jadwal latihan
	2	Pengguna memilih hari latihan.
	3	Sistem berhasil menampilkan latihan n-back.
Extensions	3.1	Sistem gagal menampilkan latihan n-back.

3. Skenario *Use Case* Latihan N-Back

Skenario *Use Case* Latihan N-Back akan dijelaskan pada Tabel 3.9 berikut ini:

Tabel 3.9 Skenario *Use Case* Latihan N-Back

Use case name	**Latihan N-Back**
Related Requirements	Tampil Latihan
Goal In Context	Pengguna melakukan latihan n-back.
Preconditions	Pengguna telah memilih hari latihan.
Successful End Condition	Sistem meng-*update* data status di *database*.
Failed End Condition	Sistem tidak meng-*update* data status di *database*.
Primary Actors	Pengguna

Use case name	**Latihan N-Back**	
Secondary Actors	-	
Trigger	Pengguna menekan tombol “Mulai”.	
Main Flow	*Step*	*Action*
	1	Pengguna menekan tombol “Mulai”.
	2	Sistem menampilkan stimulus sesuai dengan materi.
	3	Pengguna menekan tombol “Sama”.
	4	Sistem mengeluarkan bunyi “beep”.
	5	Sistem menghitung skor latihan.
	6	Sistem menampilkan keterangan lulus
	7	Sistem meng-*update* data status di *database*.
Extensions	*Step*	*Branching Action*
	4.1	Sistem mengeluarkan getaran.
	6.1	Sistem menampilkan keterangan tidak lulus.
	7.1	Sistem tidak meng-*update* data status di *database*.

4. Skenario *Use Case* Tampil Tes

Adapun Skenario *Use Case* Tampil Tes akan dijelaskan pada Tabel 3.10 berikut ini:

Tabel 3.10 Skenario *Use Case* Tampil Tes

Use case name	**Tampil Tes**
Related Requirements	Tampil Latihan
Goal In Context	Pengguna ingin menampilkan tes daya ingat.
Preconditions	Sistem menampilkan latihan.
Successful End Condition	Sistem berhasil menampilkan tes daya ingat.
Failed End Condition	Sistem gagal menampilkan tes daya ingat.
Primary Actors	Pengguna

Use case name	**Tampil Tes**	
Secondary Actors	-	
Trigger	Pengguna memilih hari latihan.	
Main Flow	*Step*	*Action*
	1	Sistem menampilkan jadwal latihan.
	2	Pengguna memilih latihan.
	3	Sistem menampilkan pesan apakah pengguna mau melakukan tes.
	4	Sistem berhasil menampilkan tes daya ingat.
Extensions	*Step*	*Branching Action*
	4.1	Sistem gagal menampilkan tes daya ingat.

5. Skenario *Use Case* Tes Daya Ingat

Adapun Skenario *Use Case* Tes Daya Ingat akan dijelaskan pada Tabel 3.11 berikut ini:

Tabel 3.11 Skenario *Use Case* Tes Daya Ingat

Use case name	**Tes Daya Ingat**	
Related Requirements	Tampil Tes	
Goal In Context	Pengguna melakukan tes daya ingat.	
Preconditions	Status latihan yaitu “Tes”	
Successful End Condition	Sistem meng-*update* data pengujian *database*.	
Failed End Condition	-	
Primary Actors	Pengguna	
Secondary Actors	-	
Trigger	Pengguna menekan tombol “Mulai”.	
Main Flow	*Step*	*Action*
	1	Pengguna menekan tombol “Mulai”.
	2	Sistem menampilkan kalimat.
	3	Pengguna memasukkan kata terakhir dari

Use case name	**Tes Daya Ingat**	
		kalimat yang ditampilkan.
	4	Sistem menghitung tingkat keberhasilan bagian yang telah diselesaikan.
	5	Sistem menampilkan skor.
	6	Sistem meng-*update* data pengujian *database*.

6. Skenario *Use Case* Tampil Daftar Tes

Skenario *Use Case* Tampil Daftar Tes akan dijelaskan pada Tabel 3.12 berikut ini:

Tabel 3.12 Skenario *Use Case* Tampil Daftar Tes

Use case name	**Tampil Daftar Tes**	
Related Requirements	Tampil Hasil Tes	
Goal In Context	Pengguna ingin menampilkan daftar tes.	
Preconditions	Sistem menampilkan menu utama.	
Successful End Condition	Sistem berhasil menampilkan daftar tes.	
Failed End Condition	Sistem gagal menampilkan daftar tes	
Primary Actors	Pengguna	
Secondary Actors	-	
Trigger	Pengguna memilih menu “Hasil Tes”.	
Main Flow	*Step*	*Action*
	1	Sistem menampilkan menu utama.
	2	Pengguna memilih menu “Hasil Tes”.
	3	Sistem berhasil menampilkan daftar tes.
Extensions	*Step*	*Branching Action*
	3.1	Sistem gagal menampilkan daftar tes.

7. Skenario *Use Case* Tampil Hasil Tes

Skenario *Use Case* Tampil Hasil Tes akan dijelaskan pada Tabel 3.13 berikut ini:

Tabel 3.13 Skenario *Use Case* Tampil Hasil Tes

Use case name	**Tampil Hasil Tes**	
Related Requirements	Tampil Daftar Tes	
Goal In Context	Pengguna ingin menampilkan hasil tes.	
Preconditions	Sistem menampilkan daftar tes.	
Successful End Condition	Sistem berhasil menampilkan hasil tes.	
Failed End Condition	Sistem gagal menampilkan hasil tes.	
Primary Actors	Pengguna	
Secondary Actors	-	
Trigger	Pengguna memilih hasil tes yang akan ditampilkan.	
Main Flow	*Step*	*Action*
	1	Sistem menampilkan daftar tes.
	2	Pengguna memilih salah satu tes.
	3	Sistem berhasil menampilkan hasil tes yang dipilih.
Extensions	*Step*	*Branching Action*
	3.1	Sistem gagal menampilkan hasil tes yang dipilih.

8. Skenario *Use Case* Tampil Daftar Penjelasan

Skenario *Use Case* Tampil Daftar Penjelasan akan dijelaskan pada Tabel 3.14 berikut ini:

Tabel 3.14 Skenario *Use Case* Tampil Daftar Penjelasan

Use case name	**Tampil Daftar Penjelasan**
Related Requirements	Tampil Penjelasan
Goal In Context	Pengguna ingin menampilkan daftar penjelasan.

Use case name	**Tampil Daftar Penjelasan**	
Preconditions	Sistem menampilkan menu utama.	
Successful End Condition	Sistem berhasil menampilkan daftar penjelasan.	
Failed End Condition	-	
Primary Actors	Pengguna	
Secondary Actors	-	
Trigger	Pengguna memilih menu “Penjelasan”.	
Main Flow	*Step*	*Action*
	1	Sistem menampilkan menu utama.
	2	Pengguna memilih menu “Penjelasan”.
	3	Sistem berhasil menampilkan daftar penjelasan.

9. Skenario *Use Case* Tampil Penjelasan

Skenario *Use Case* Tampil Penjelasan akan dijelaskan pada Tabel 3.15 berikut ini:

Tabel 3.15 Skenario *Use Case* Tampil Penjelasan

Use case name	**Tampil Penjelasan**	
Related Requirements	Tampil Daftar Penjelasan	
Goal In Context	Pengguna ingin menampilkan penjelasan.	
Preconditions	Sistem menampilkan daftar penjelasan.	
Successful End Condition	Sistem berhasil menampilkan penjelasan.	
Failed End Condition	-	
Primary Actors	Pengguna	
Secondary Actors	-	
Trigger	Pengguna memilih penjelasan yang akan ditampilkan.	
Main Flow	*Step*	*Action*
	1	Sistem menampilkan daftar penjelasan
	2	Pengguna memilih penjelasan.

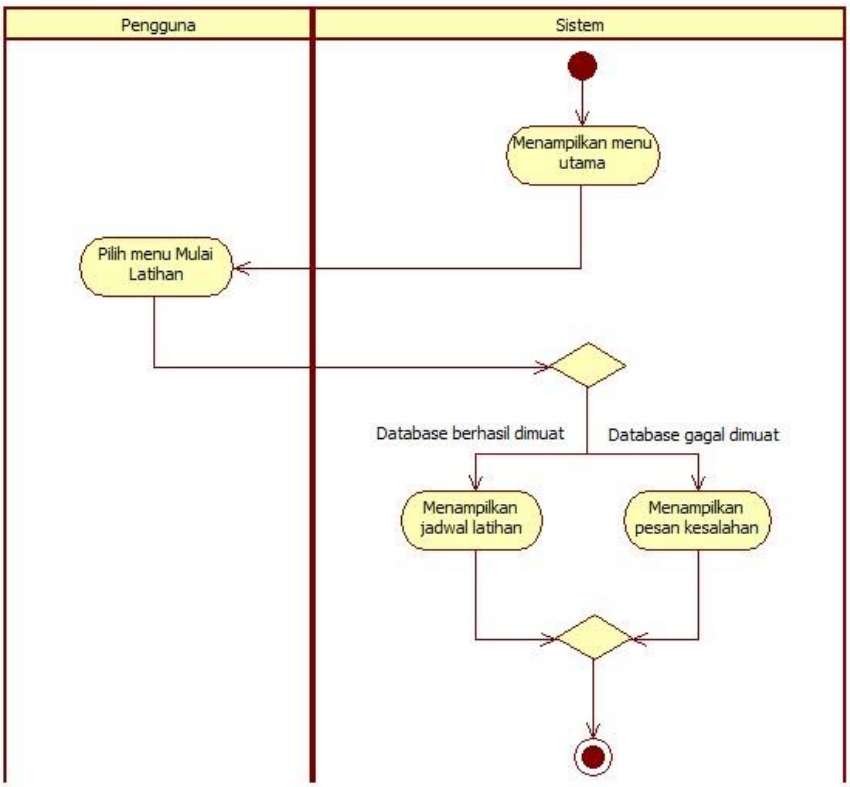
Use case name	**Tampil Penjelasan**	
	3	Sistem berhasil menampilkan penjelasan yang dipilih.

3.1.6.2 *Activity Diagram*

Activity diagram merupakan diagram yang memodelkan aliran kerja atau *workflow* dari urutan aktivitas dalam suatu proses yang mengacu pada *use case diagram* yang ada.

1. *Activity Diagram* Tampil Jadwal Latihan

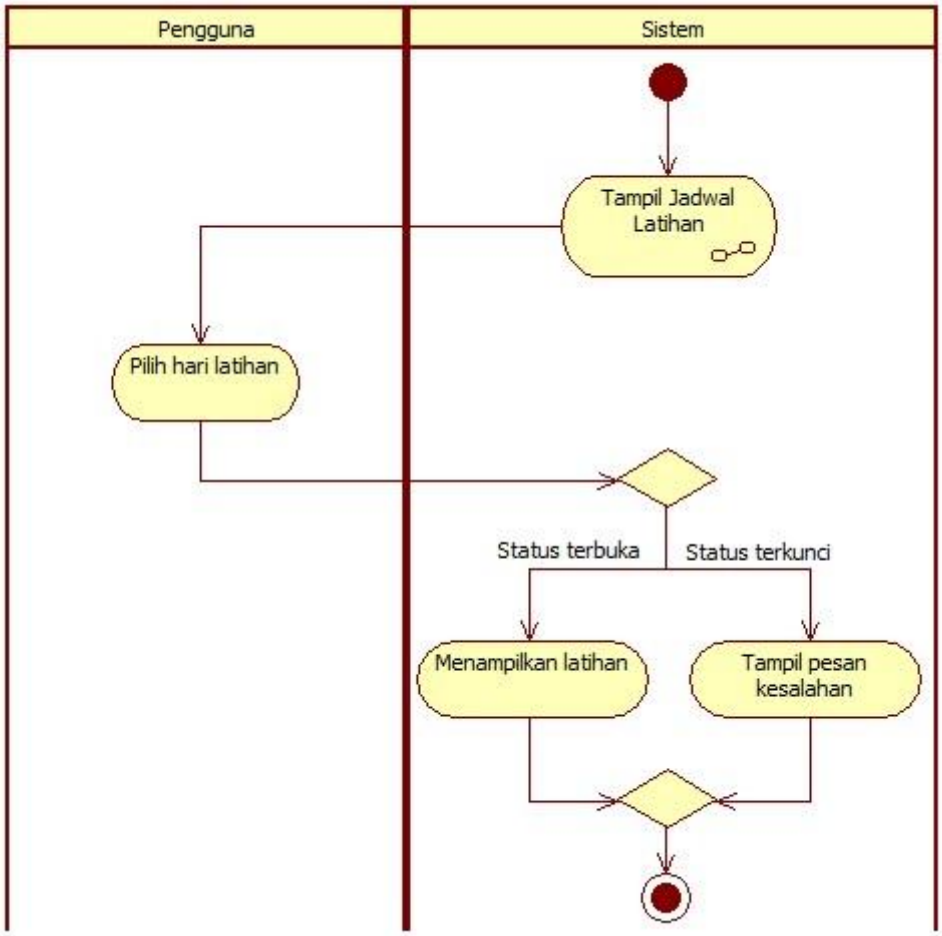
Activity diagram tampil jadwal latihan menjelaskan aliran kerja aktor pengguna pada saat menampilkan jadwal latihan. *Activity diagram* tampil jadwal latihan dapat dilihat pada Gambar 3.7.



Gambar 3.7 *Activity Diagram* Tampil Jadwal Latihan

2. *Activity Diagram* Tampil Latihan

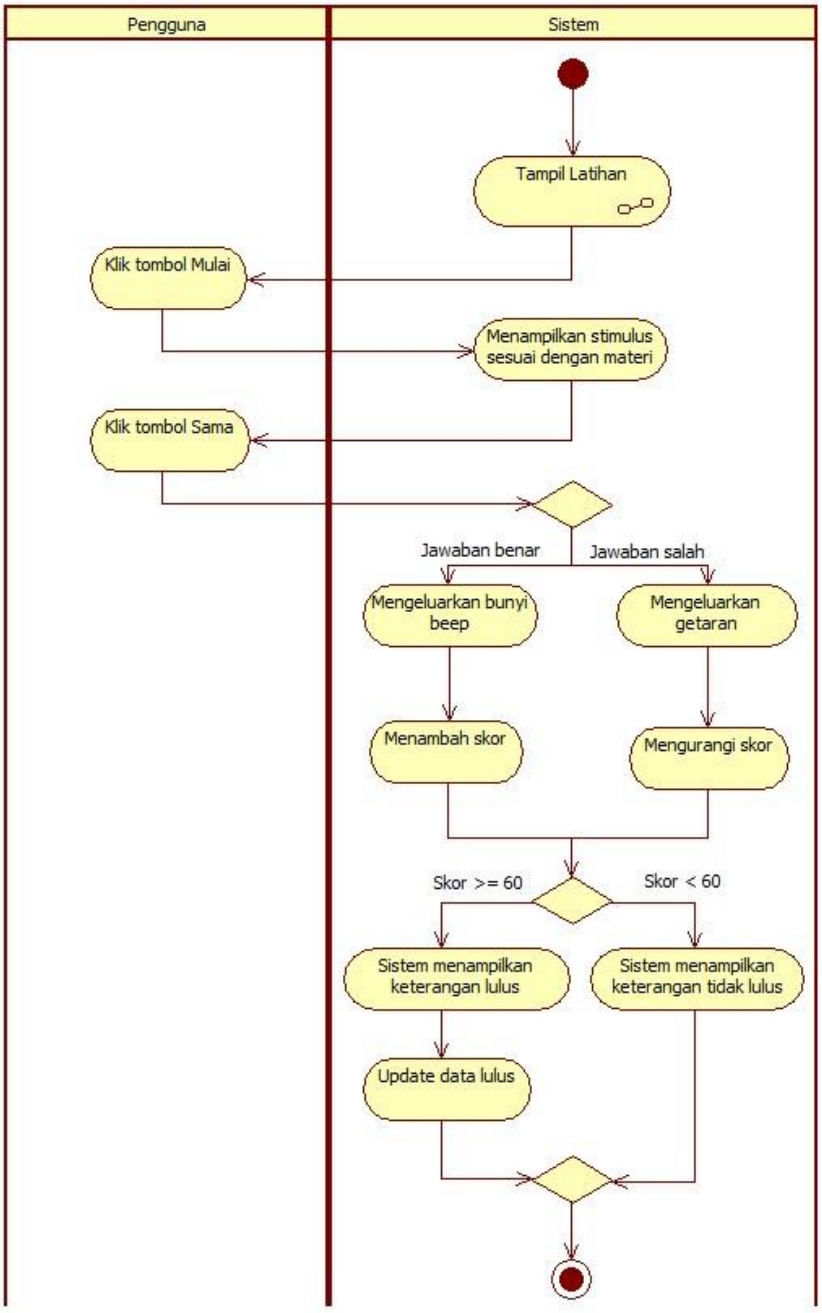
Activity diagram tampil latihan menjelaskan aliran kerja aktor pengguna pada saat menampilkan latihan n-back. *Activity diagram* tampil latihan dapat dilihat pada Gambar 3.8.



Gambar 3.8 *Activity Diagram* Tampil Latihan

3. *Activity Diagram* Latihan N-Back

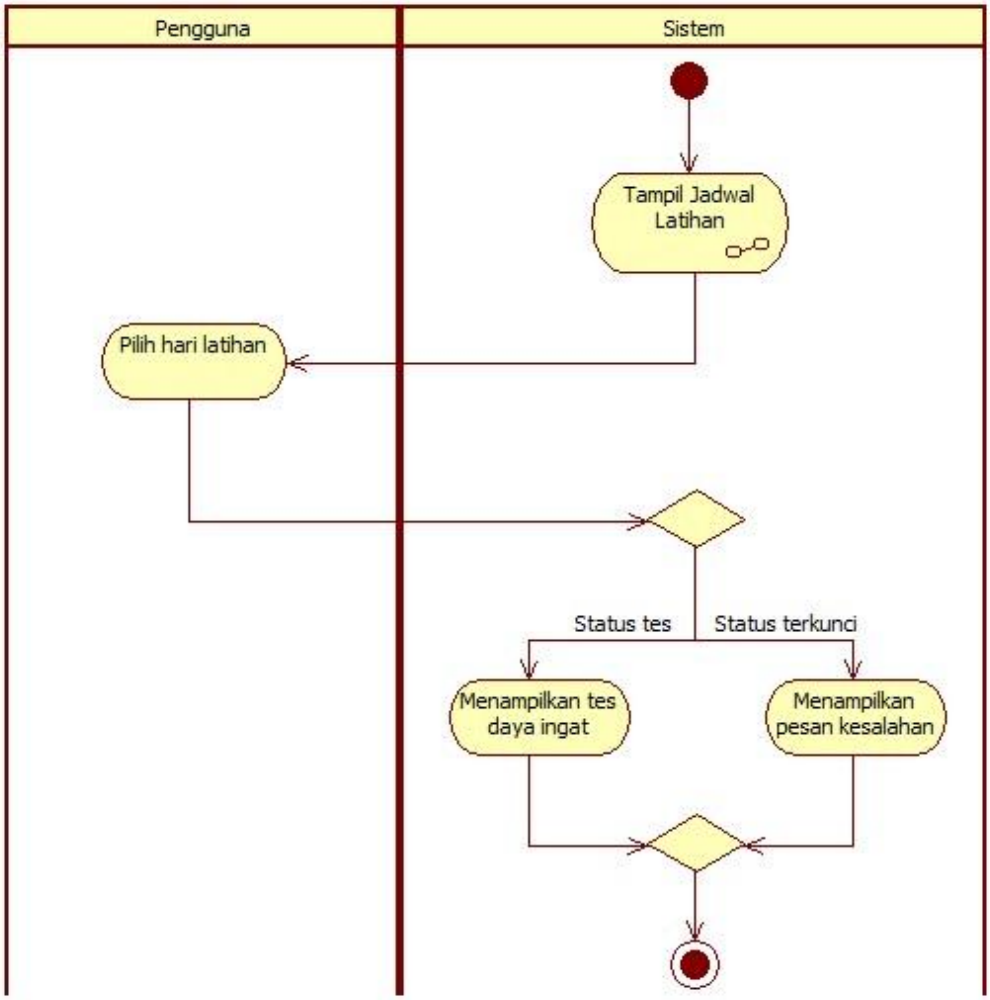
Activity diagram latihan n-back menjelaskan aliran kerja aktor pengguna pada saat mengerjakan latihan n-back yang dipilih. *Activity diagram* latihan n-back dapat dilihat pada Gambar 3.9.



Gambar 3.9 *Activity Diagram* Latihan N-Back

4. *Activity Diagram* Tampil Tes

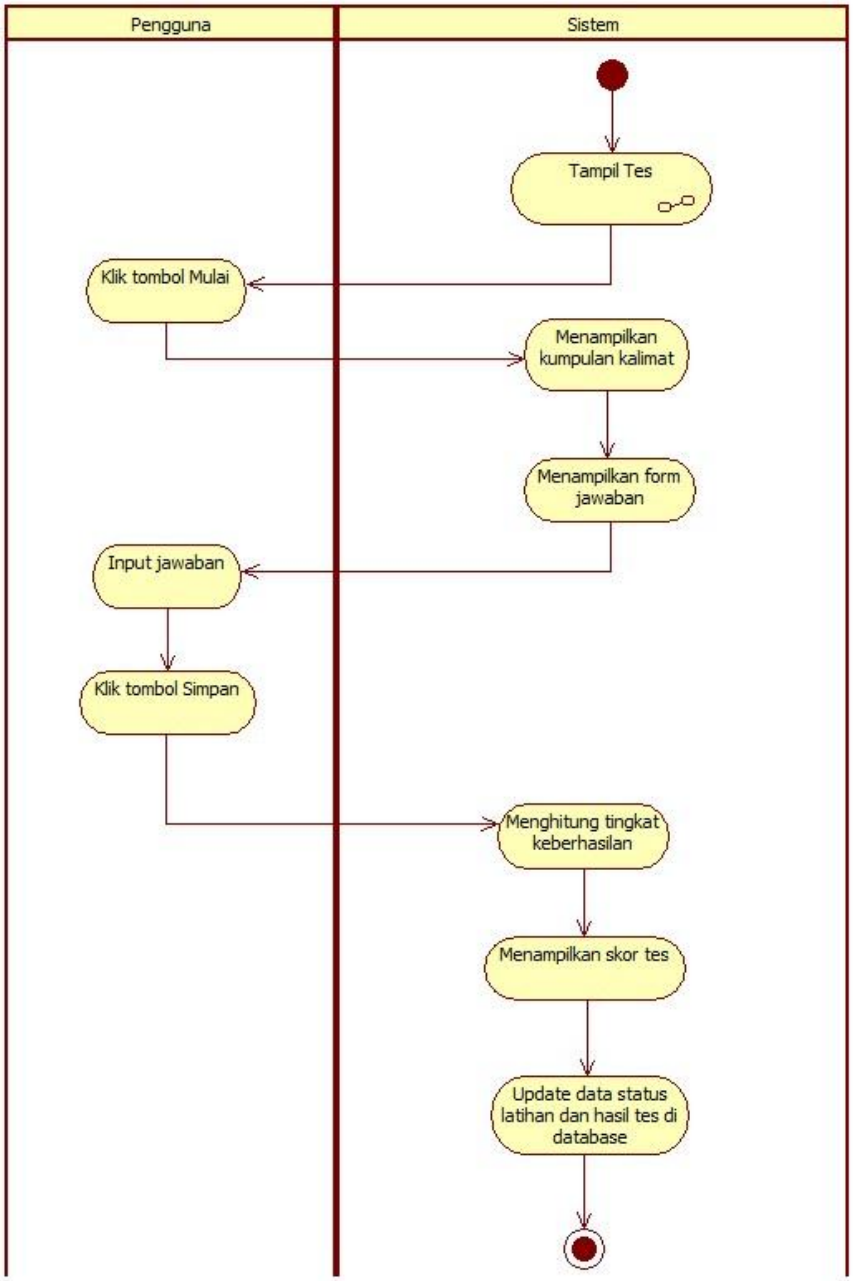
Activity diagram tampil tes menjelaskan aliran kerja sistem pada saat menampilkan tes daya ingat. *Activity diagram* tampil tes dapat dilihat pada Gambar 3.10.



Gambar 3.10 *Activitty Diagram* Tampil Tes

5. *Activity Diagram* Tes Daya Ingat

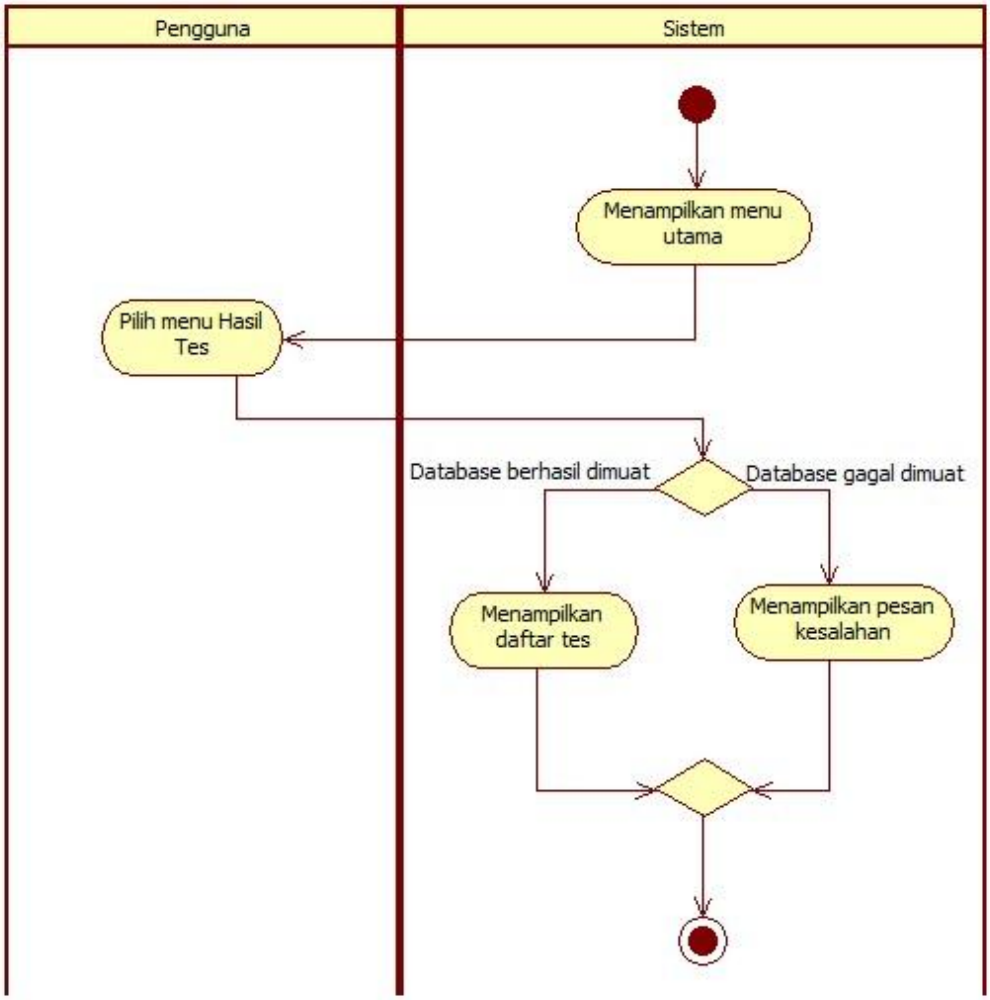
Activity diagram tes daya ingat menjelaskan aliran kerja sistem pada saat mengerjakan tes daya ingat. *Activity diagram* tes daya ingat dapat dilihat pada Gambar 3.11.



Gambar 3.11 *Activity Diagram* Tes Daya Ingat

6. *Activity Diagram* Tampil Daftar Tes

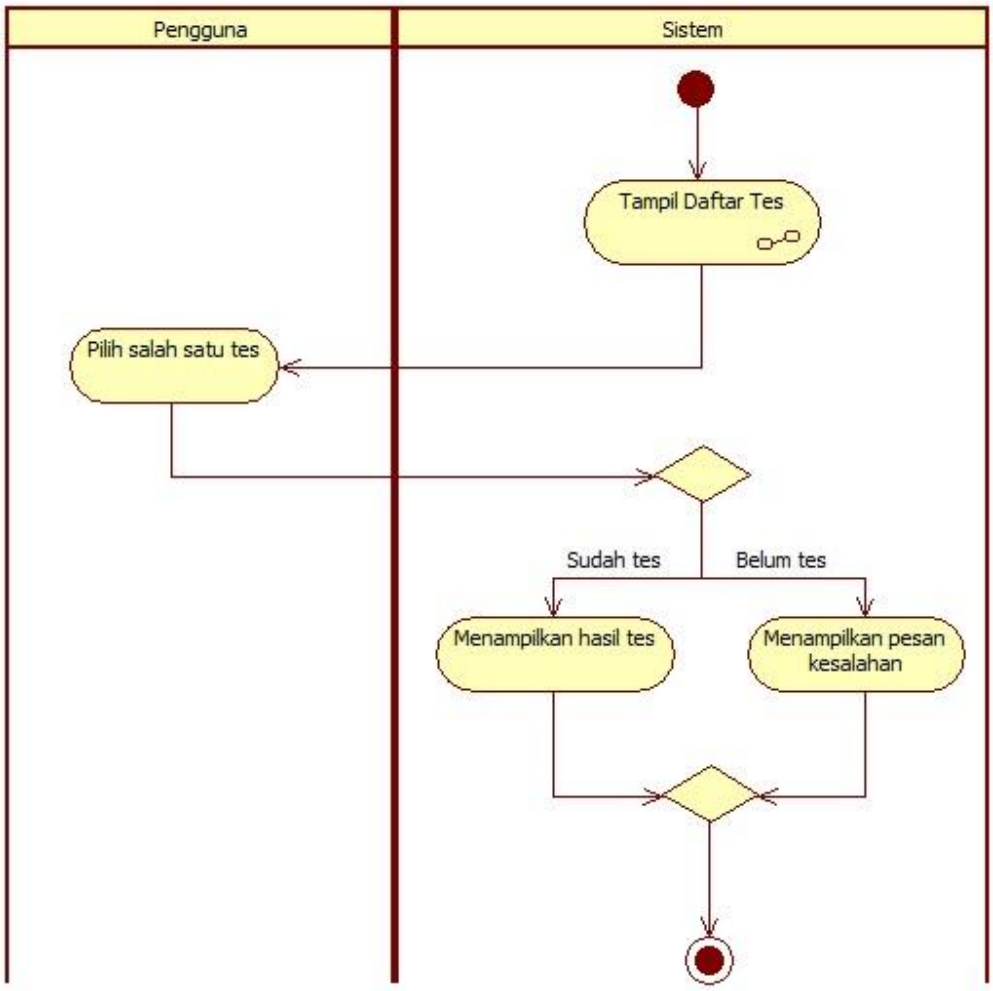
Activity diagram tampil daftar tes menjelaskan aliran kerja aktor pengguna pada saat menampilkan daftar tes. *Activity diagram* tampil daftar tes dapat dilihat pada Gambar 3.12.



Gambar 3.12 *Activity Diagram* Tampil Daftar Tes

7. *Activity Diagram* Tampil Hasil Tes

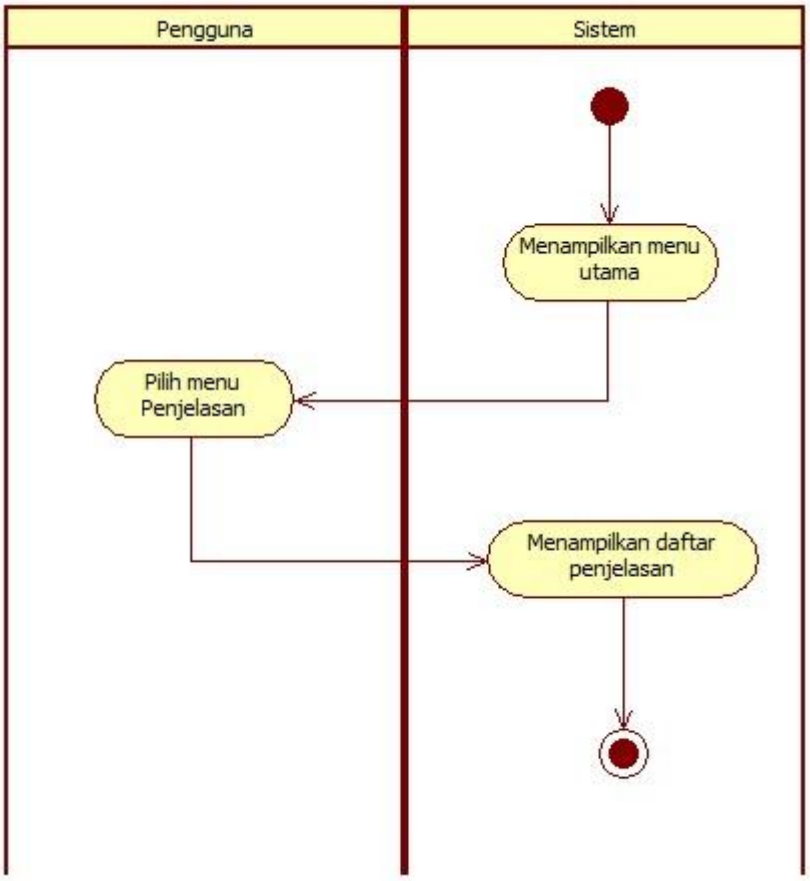
Activity diagram tampil hasil tes menjelaskan aliran kerja aktor pengguna pada saat menampilkan hasil tes. *Activity diagram* tampil hasil tes dapat dilihat pada Gambar 3.13.



Gambar 3.13 *Activity Diagram* Tampil Hasil Tes

8. *Activity Diagram* Tampil Daftar Penjelasan

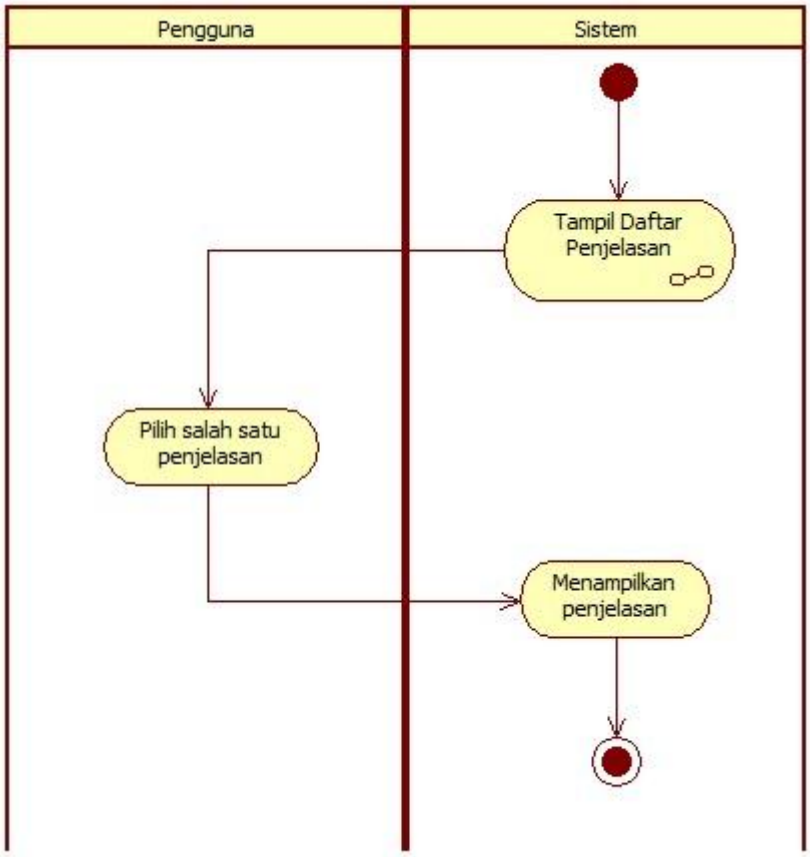
Activity diagram tampil daftar penjelasan menjelaskan aliran kerja aktor pengguna pada saat menampilkan daftar penjelasan. *Activity diagram* tampil daftar penjelasan dapat dilihat pada Gambar 3.14.



Gambar 3.14 *Activity Diagram* Tampil Daftar Penjelasan

9. *Activity Diagram* Tampil Penjelasan

Activity diagram tampil penjelasan menjelaskan aliran kerja aktor pengguna pada saat menampilkan penjelasan yang dipilih. *Activity diagram* tampil penjelasan dapat dilihat pada Gambar 3.15.



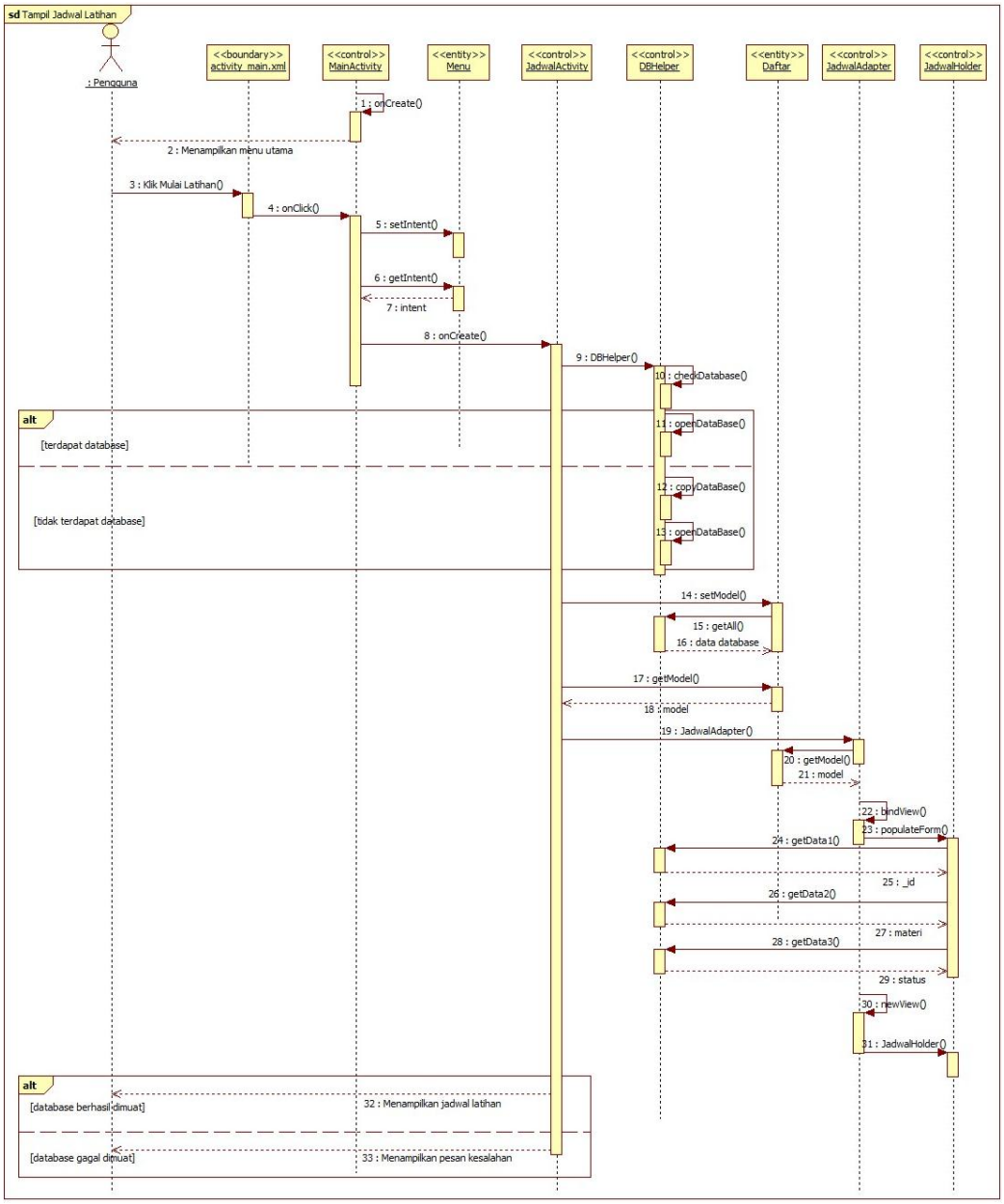
Gambar 3.15 *Activity Diagram* Tampil Penjelasan

3.1.6.3 *Sequence Diagram*

Sequence diagram merupakan gambaran interaksi antar masing-masing objek pada setiap *use case* dalam urutan waktu. Interaksi ini berupa pengiriman serangkaian data antar objek – objek yang saling berinteraksi.

1. *Sequence Diagram* Tampil Jadwal Latihan

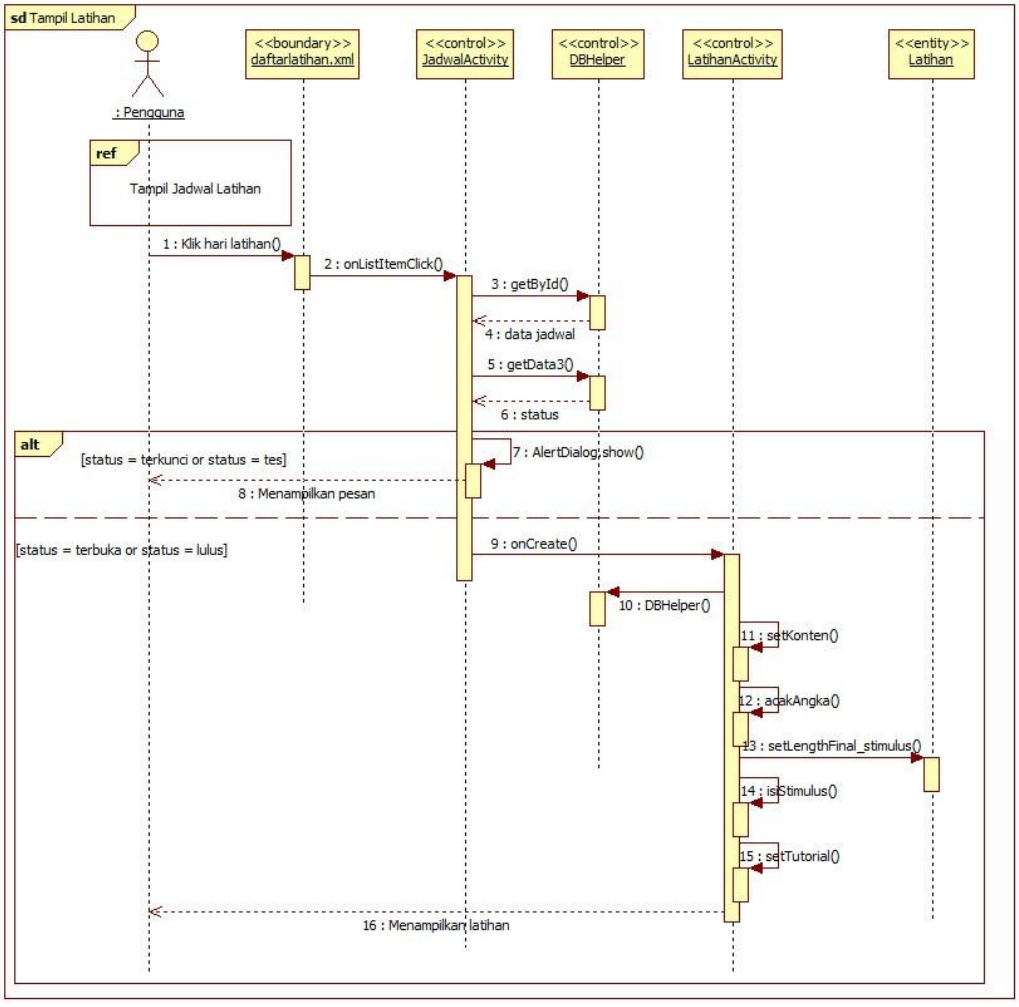
Sequence diagram tampil jadwal latihan dapat dilihat pada Gambar 3.16.



Gambar 3.16 *Sequence Diagram* Tampil Jadwal Latihan

2. *Sequence Diagram* Tampil Latihan

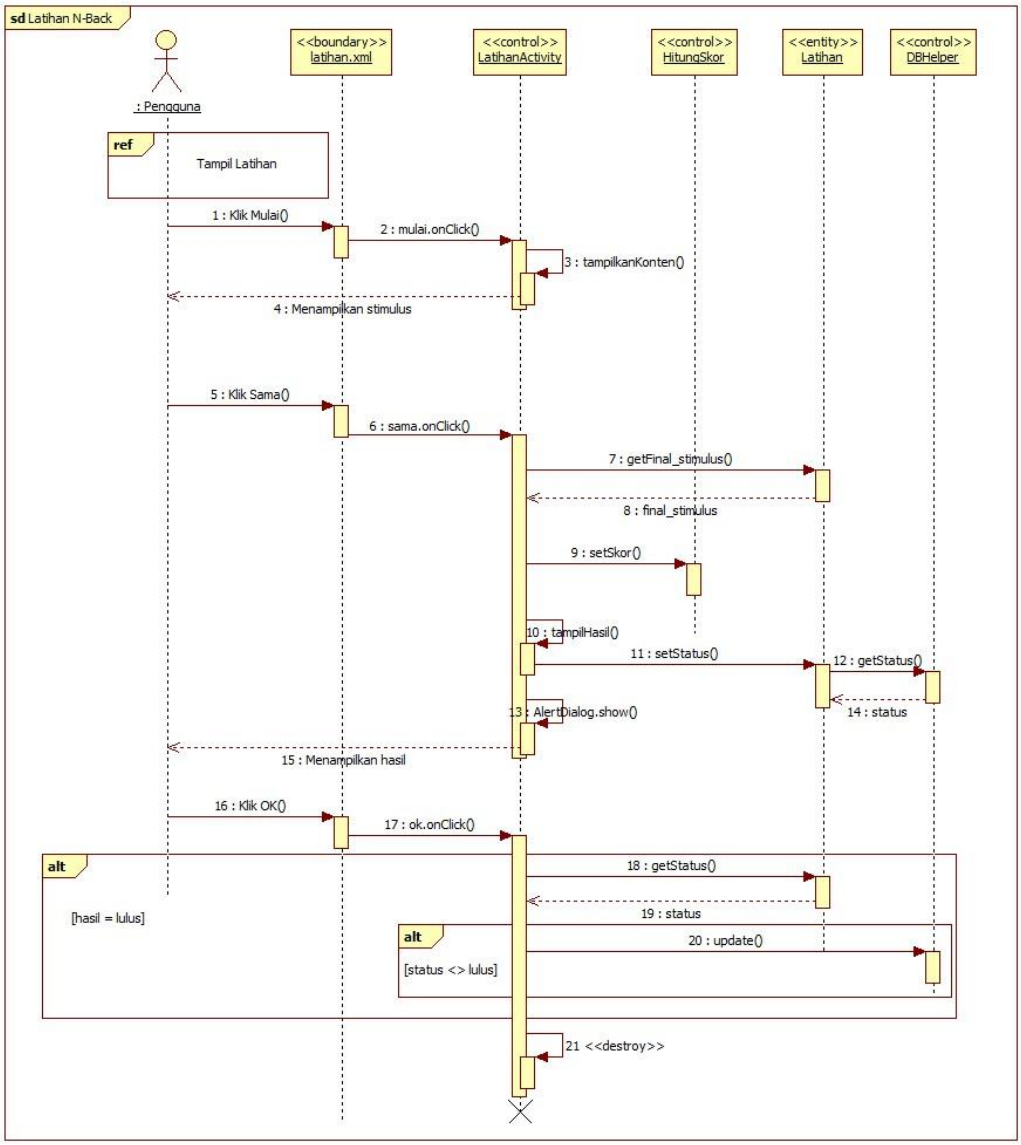
Sequence diagram tampil latihan menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses tampil latihan. *Sequence diagram* tampil latihan dapat dilihat pada Gambar 3.17.



Gambar 3.17 *Sequence Diagram* Tampil Latihan

3. *Sequence Diagram* Latihan N-Back

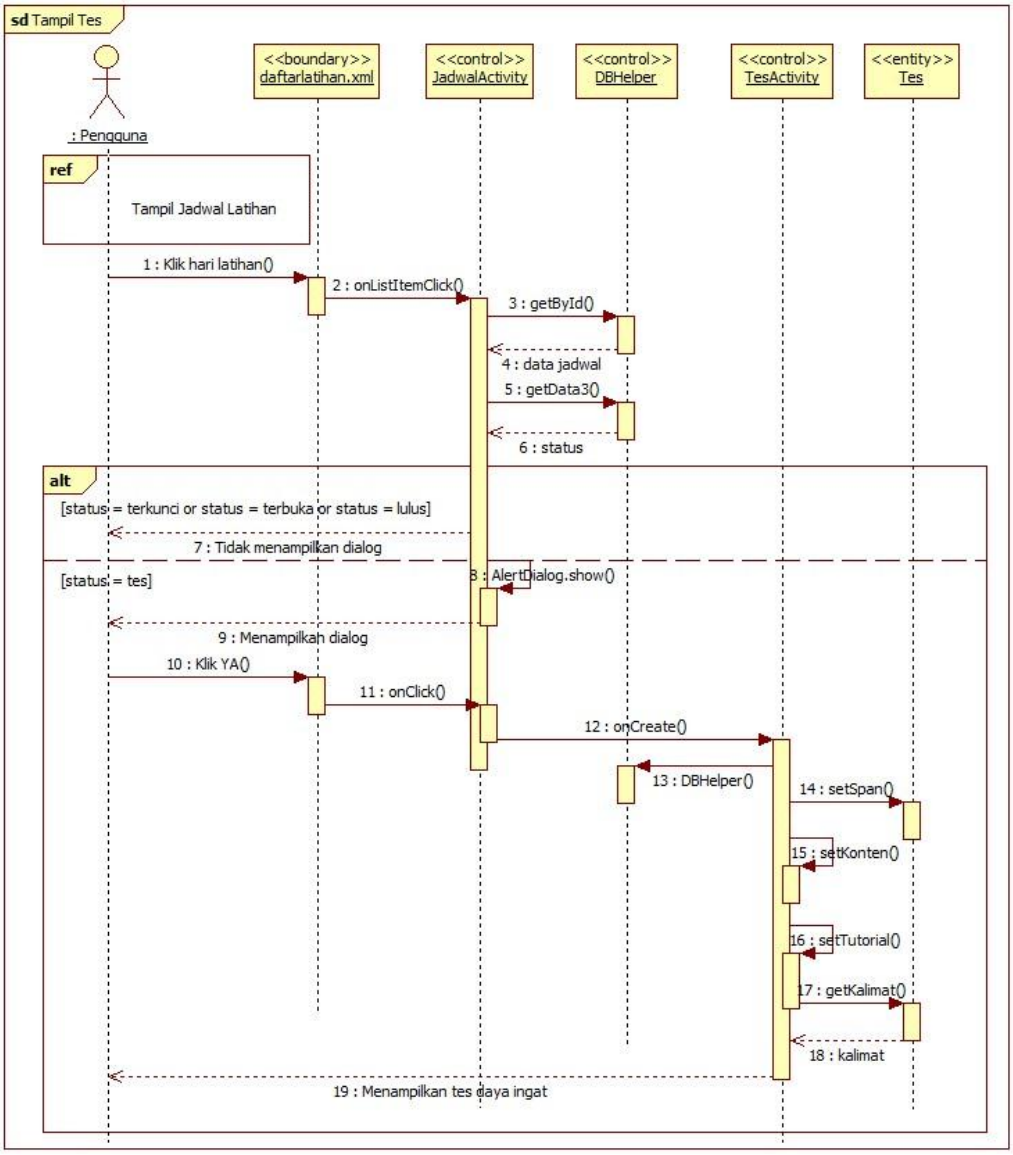
Sequence diagram latihan n-back menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses latihan n-back. *Sequence diagram* latihan n-back dapat dilihat pada Gambar 3.18.



Gambar 3.18 *Sequence Diagram* Latihan N-Back

4. *Sequence Diagram* Tampil Tes

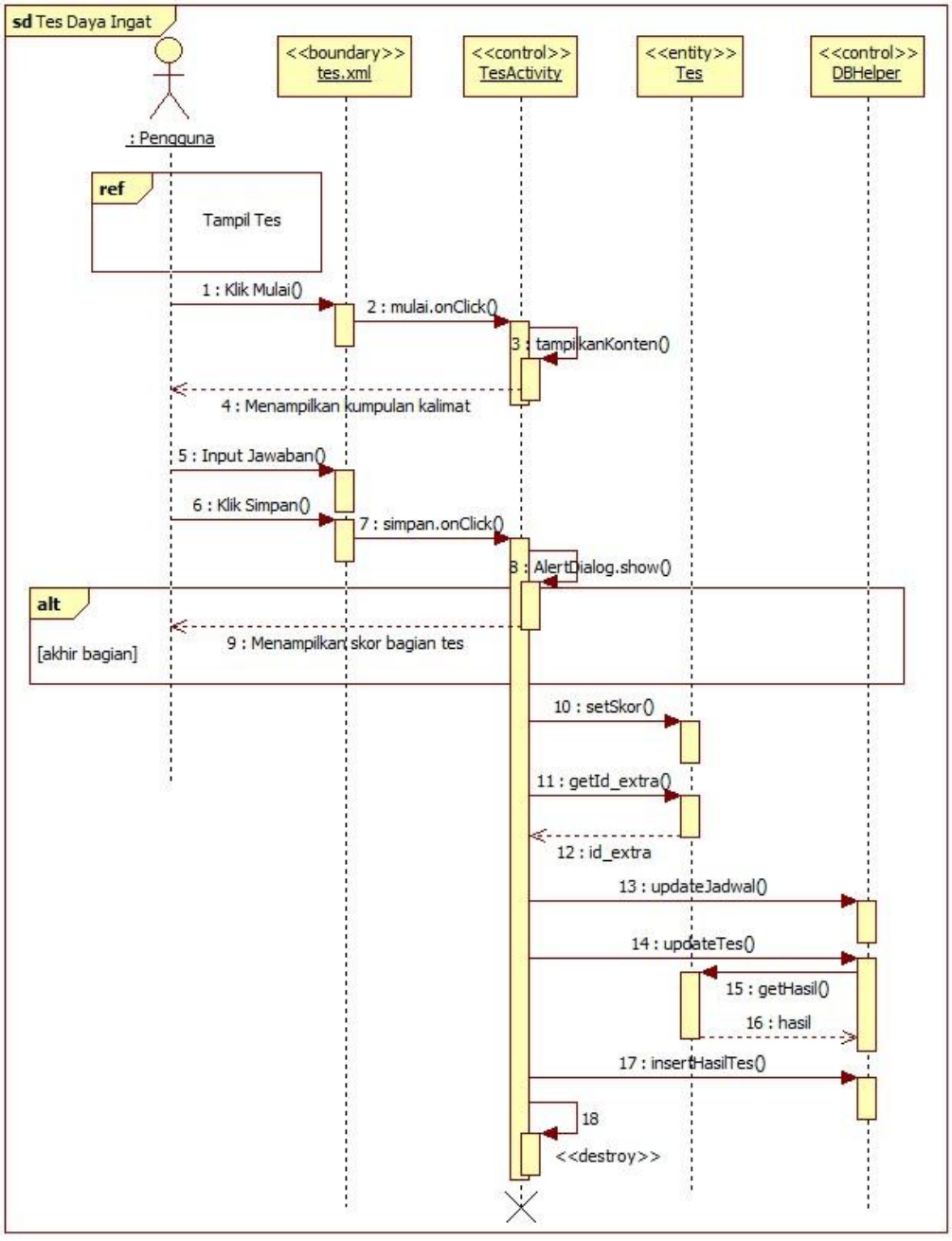
Sequence diagram tampil tes menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses tampil tes. *Sequence diagram* tampil tes dapat dilihat pada Gambar 3.19.



Gambar 3.19 *Sequence Diagram* Tampil Tes

5. *Sequence Diagram* Tes Daya Ingat

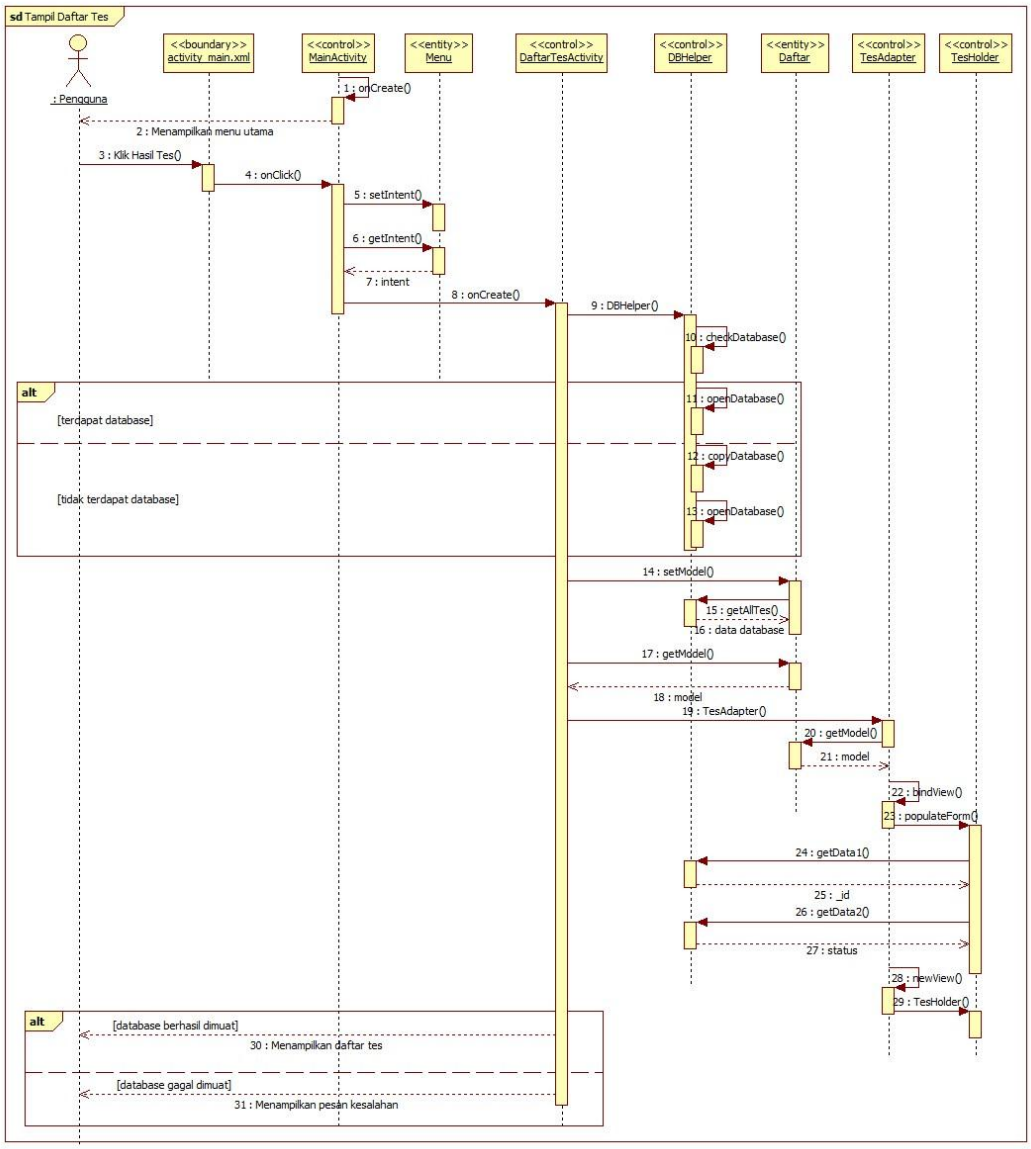
Sequence diagram tes daya ingat menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses tes daya ingat. *Sequence diagram* tes daya ingat dapat dilihat pada Gambar 3.20.



Gambar 3.20 *Sequence Diagram* Tes Daya Ingat

6. *Sequence Diagram* Tampil Daftar Tes

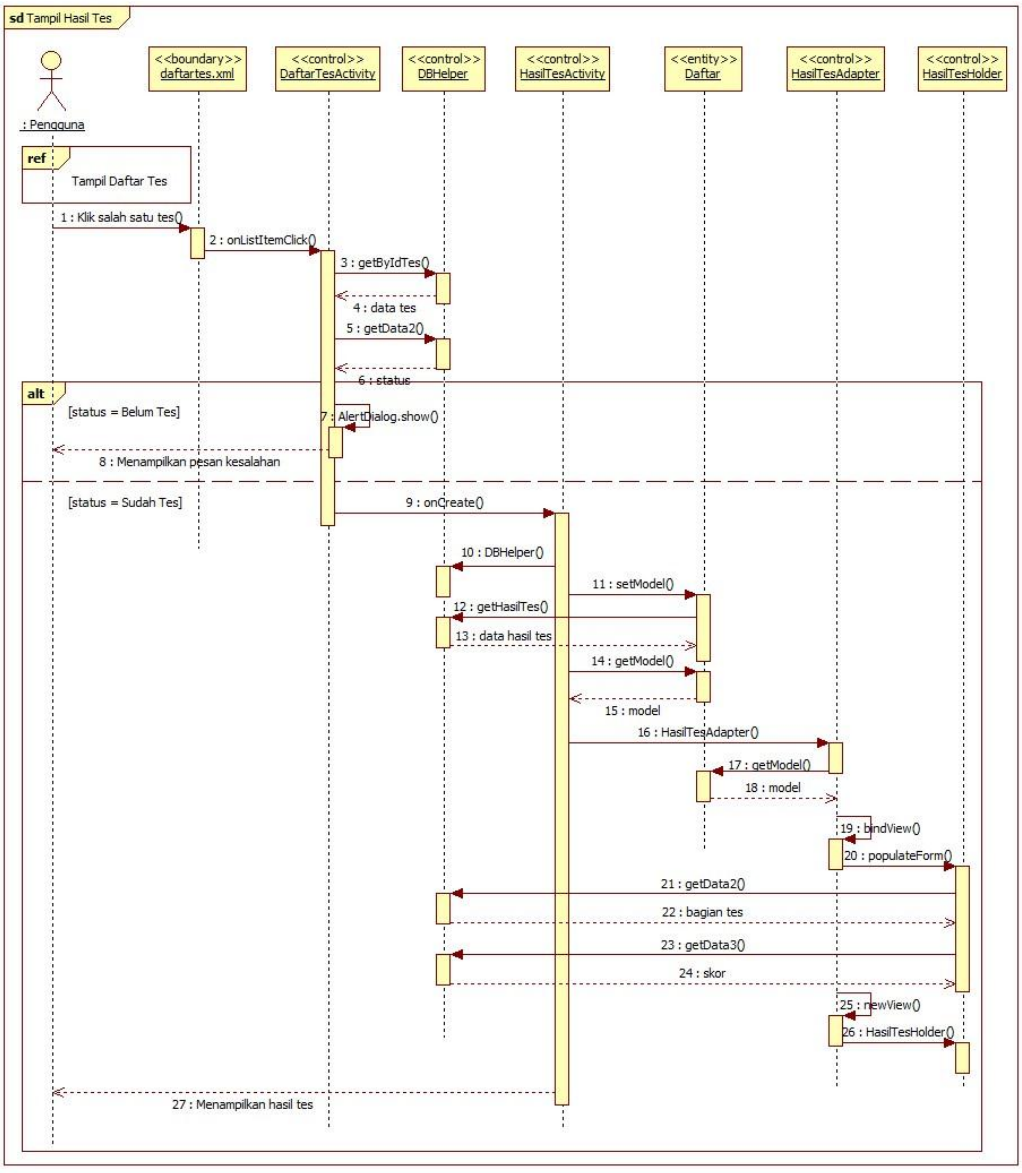
Sequence diagram tampil daftar tes menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses tampil daftar tes. *Sequence diagram* tampil daftar tes dapat dilihat pada Gambar 3.21.



Gambar 3.21 *Sequence Diagram* Tampil Daftar Tes

7. *Sequence Diagram* Tampil Hasil Tes

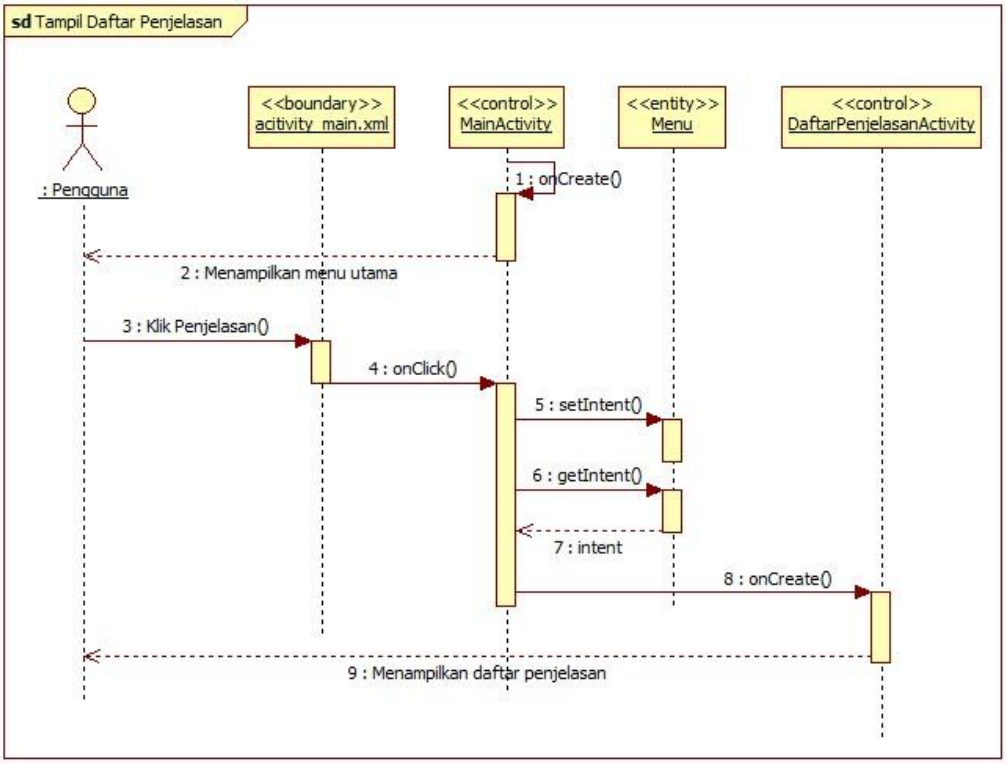
Sequence diagram tampil hasil tes menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses tampil hasil tes. *Sequence diagram* tampil hasil tes dapat dilihat pada Gambar 3.22.



Gambar 3.22 *Sequence Diagram* Tampil Hasil Tes

8. *Sequence Diagram* Tampil Daftar Penjelasan

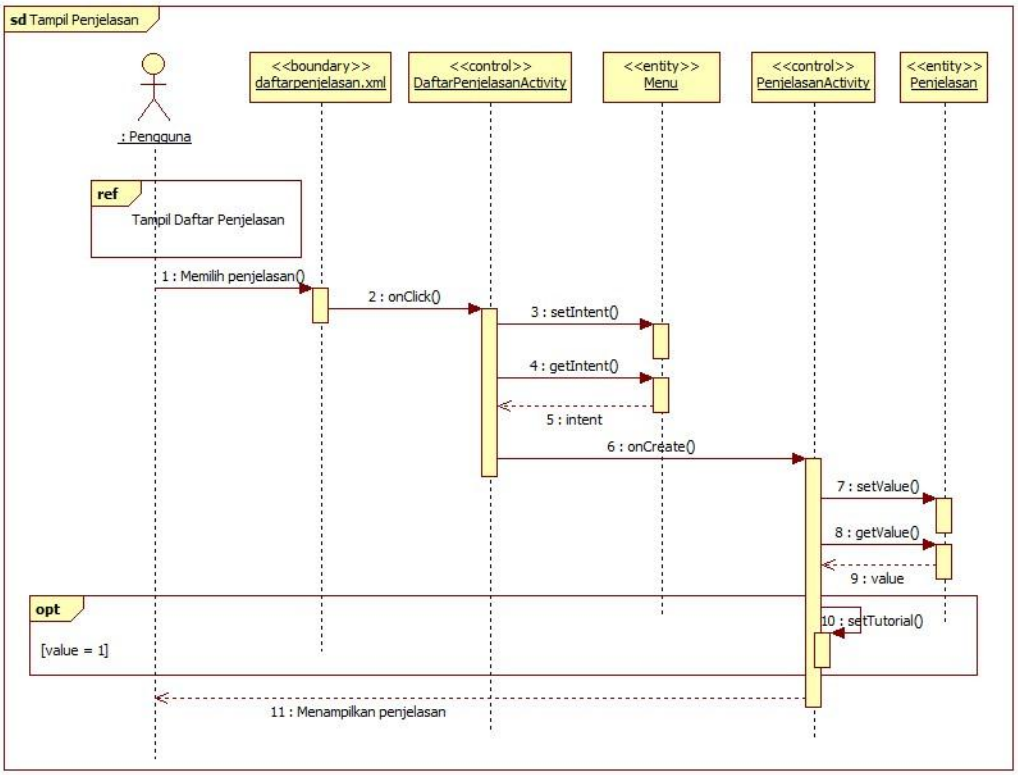
Sequence diagram tampil daftar penjelasan menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses tampil daftar penjelasan. *Sequence diagram* tampil daftar penjelasan dapat dilihat pada Gambar 3.23.



Gambar 3.23 *Sequence Diagram* Tampil Daftar Penjelasan

9. *Sequence Diagram* Tampil Penjelasan

Sequence diagram tampil penjelasan menunjukkan rangkaian pesan yang dikirim antara objek pengguna dengan objek lainnya dalam proses tampil penjelasan. *Sequence diagram* tampil penjelasan dapat dilihat pada Gambar 3.24.



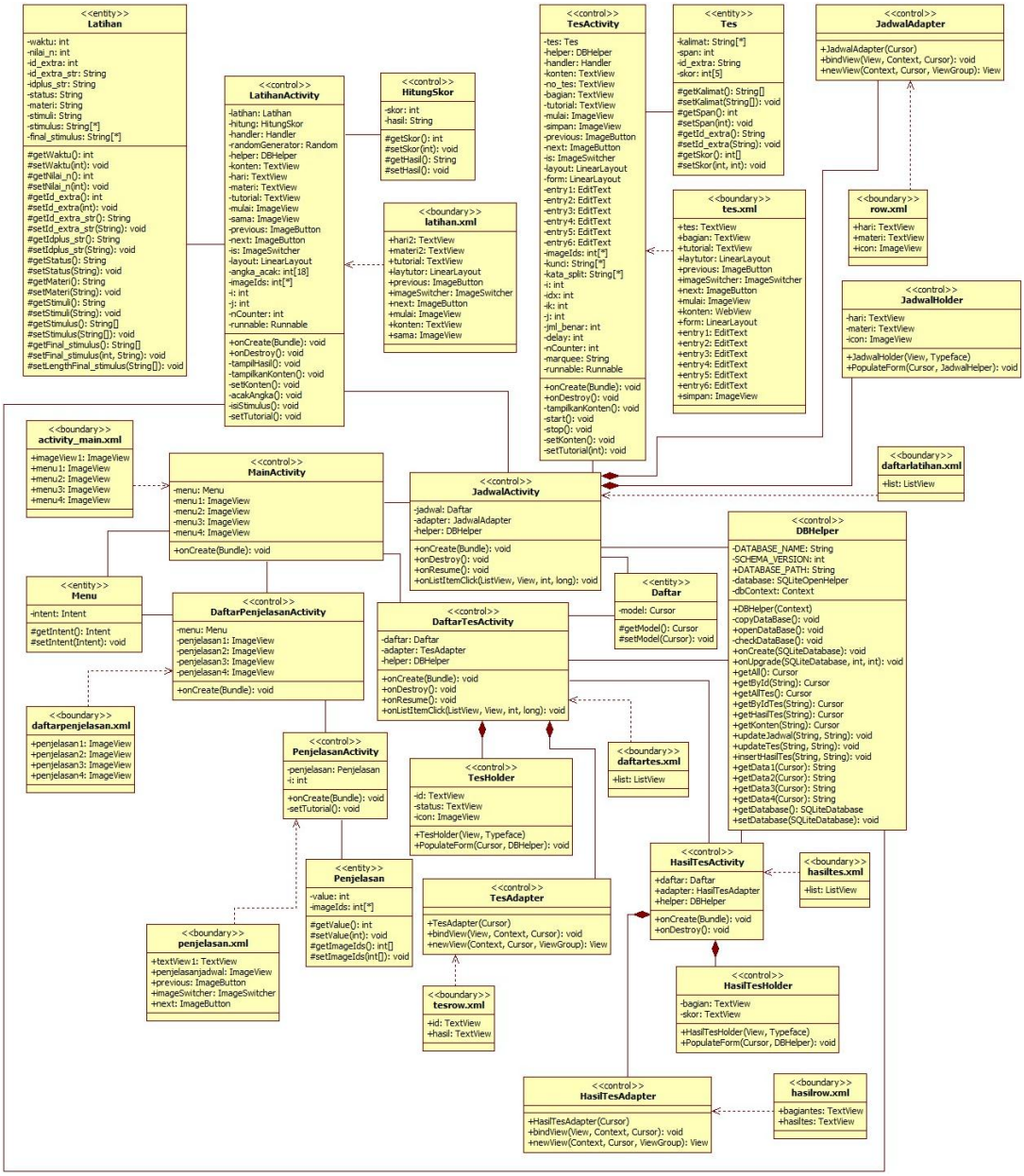
Gambar 3.24 *Sequence Diagram* Tampil Penjelasan

3.1.6.4 Deskripsi Kelas

Deskripsi kelas berisi pemaparan *class diagram* beserta daftar kelas yang dikelompokkan berdasarkan jenis kelas, dan spesifikasi masing-masing kelas.

3.1.6.4.1 *Class Diagram*

Adapun gambaran *class diagram* dari aplikasi *mobile* "Latihan Daya Ingat" yang dibangun dapat dilihat pada Gambar 3.25.



Gambar 3.25 *Class Diagram* Aplikasi "Latihan Daya Ingat"

3.1.6.4.2 Daftar Kelas

Daftar kelas berisi jenis dari setiap kelas yang terdapat pada *class diagram* aplikasi *mobile* "Latihan Daya Ingat" (Gambar 3.25). *Class diagram* aplikasi *mobile* "Latihan Daya Ingat" yang dibangun terdiri dari tiga jenis kelas, yaitu *class entity*, *class boundary*, dan *class control*. Adapun daftar setiap kelas yang terdapat pada *class diagram* aplikasi *mobile* "Latihan Daya Ingat" berdasarkan jenisnya dapat dilihat pada Tabel 3.16.

Tabel 3.16 Daftar Kelas

No.	Nama Kelas	Jenis
1	activity_main.xml	*Boundary*
2	daftarlatihan.xml	*Boundary*
3	daftarpenjelasan.xml	*Boundary*
4	daftartes.xml	*Boundary*
5	hasilrow.xml	*Boundary*
6	hasiltes.xml	*Boundary*
7	latihan.xml	*Boundary*
8	penjelasan.xml	*Boundary*
9	row.xml	*Boundary*
10	tes.xml	*Boundary*
11	tesrow.xml	*Boundary*
12	DaftarPenjelasanActivity	*Control*
13	DaftarTesActivity	*Control*
14	DBHelper	*Control*
15	HasilTesActivity	*Control*
16	HasilTesAdapter	*Control*
17	HasilTesHolder	*Control*
18	HitungSkor	*Control*
19	JadwalActivity	*Control*
20	JadwalAdapter	*Control*

No.	Nama Kelas	Jenis
21	JadwalHolder	*Control*
22	LatihanActivity	*Control*
23	MainActivity	*Control*
24	PenjelasanActivity	*Control*
25	TesActivity	*Control*
26	TesAdapter	*Control*
27	TesHolder	*Control*
28	Daftar	*Entity*
29	Latihan	*Entity*
30	Menu	*Entity*
31	Penjelasan	*Entity*
32	Tes	*Entity*

3.1.6.4.3 Spesifikasi Kelas

Spesifikasi kelas berisi penjelasan detail atribut dan fungsi metode yang dimiliki setiap kelas. Berikut ini penjelasan dari setiap kelas yang terdapat pada Tabel 3.16.

1. Kelas activity_main.xml

Kelas activity_main.xml merupakan kelas yang digunakan untuk menampilkan menu utama aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas activity_main.xml yang dapat dilihat pada Tabel 3.17.

Tabel 3.17 Atribut kelas activity_main.xml

Nama Atribut	*Visibility*	Tipe
imageView1	public	ImageView
menu1	public	ImageView
menu2	public	ImageView
menu3	public	ImageView

Nama Atribut	*Visibility*	Tipe
menu4	public	ImageView

2. Kelas daftarlatihan.xml

Kelas daftarlatihan.xml merupakan kelas yang digunakan untuk menampilkan jadwal latihan pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas daftarlatihan.xml yang dapat dilihat pada Tabel 3.18.

Tabel 3.18 Atribut kelas daftarlatihan.xml

Nama Atribut	*Visibility*	Tipe
list	public	ListView

3. Kelas daftarpenjelasan.xml

Kelas daftarpenjelasan.xml merupakan kelas yang digunakan untuk menampilkan daftar penjelasan pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas daftarpenjelasan.xml yang dapat dilihat pada Tabel 3.19.

Tabel 3.19 Atribut kelas daftarpenjelasan.xml

Nama Atribut	*Visibility*	Tipe
penjelasan1	public	ImageView
penjelasan2	public	ImageView
penjelasan3	public	ImageView
penjelasan4	public	ImageView

4. Kelas daftartes.xml

Kelas daftartes.xml merupakan kelas yang digunakan untuk menampilkan jadwal tes daya ingat pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas daftartes.xml yang dapat dilihat pada Tabel 3.20.

Tabel 3.20 Atribut kelas daftartes.xml

Nama Atribut	*Visibility*	Tipe
list	public	ListView

5. Kelas hasilrow.xml

Kelas hasilrow.xml merupakan kelas yang digunakan untuk menampilkan detail dari hasil tes pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas hasilrow.xml yang dapat dilihat pada Tabel 3.21.

Tabel 3.21 Atribut kelas hasilrow.xml

Nama Atribut	*Visibility*	Keterangan
bagiantes	public	TextView
hasiltes	public	TextView

6. Kelas hasiltes.xml

Kelas hasiltes.xml merupakan kelas yang digunakan untuk menampilkan hasil tes daya ingat pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas hasiltes.xml yang dapat dilihat pada Tabel 3.22.

Tabel 3.22 Atribut kelas hasiltes.xml

Nama Atribut	*Visibility*	Tipe
list	public	ListView

7. Kelas latihan.xml

Kelas latihan.xml merupakan kelas yang digunakan untuk menampilkan latihan n-back pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas latihan.xml yang dapat dilihat pada Tabel 3.23.

Tabel 3.23 Atribut kelas latihan.xml

Nama Atribut	*Visibility*	Tipe
hari2	public	TextView
materi2	public	TextView
tutorial	public	TextView
laytutor	public	LinearLayout
previous	public	ImageButton
imageSwitcher	public	ImageSwitcher
next	public	ImageButton
mulai	public	ImageView
konten	public	TextView
sama	public	ImageView

8. Kelas penjelasan.xml

Kelas penjelasan.xml merupakan kelas yang digunakan untuk menampilkan penjelasan latihan pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas penjelasan.xml yang dapat dilihat pada Tabel 3.24.

Tabel 3.24 Atribut kelas penjelasan.xml

Nama Atribut	*Visibility*	Tipe
textView1	public	TextView
penjelasanjadwal	public	ImageView
previous	public	ImageButton
imageSwitcher	public	ImageSwitcher
next	public	ImageButton

9. Kelas row.xml

Kelas row.xml merupakan kelas yang digunakan untuk menampilkan detail dari jadwal latihan pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas row.xml yang dapat dilihat pada Tabel 3.25.

Tabel 3.25 Atribut kelas row.xml

Nama Atribut	*Visibility*	Keterangan
hari	public	TextView
materi	public	TextView
icon	public	ImageView

10. Kelas tes.xml

Kelas tes.xml merupakan kelas yang digunakan untuk menampilkan tes daya ingat pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas tes.xml yang dapat dilihat pada Tabel 3.26.

Tabel 3.26 Atribut kelas tes.xml

Nama Atribut	*Visibility*	Tipe
tes	public	TextView
bagian	public	TextView
tutorial	public	TextView
laytutor	public	LinearLayout
previous	public	ImageButton
imageSwitcher	public	ImageSwitcher
next	public	ImageButton
mulai	public	ImageView
konten	public	WebView
form	public	LinearLayout
entry1	public	EditText
entry2	public	EditText
entry3	public	EditText
entry4	public	EditText
entry5	public	EditText
entry6	public	EditText
simpan	public	ImageView

11. Kelas tesrow.xml

Kelas tesrow.xml merupakan kelas yang digunakan untuk menampilkan detail dari jadwal tes pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas tesrow.xml yang dapat dilihat pada Tabel 3.27.

Tabel 3.27 Atribut kelas tesrow.xml

Nama Atribut	*Visibility*	Keterangan
id	public	TextView
hasil	public	TextView

12. Kelas DaftarPenjelasanActivity

Kelas DaftarPenjelasanActivity merupakan kelas yang digunakan untuk mengontrol daftar penjelasan dari jadwal latihan pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas DaftarPenjelasanActivity yang dapat dilihat pada Tabel 3.28.

Tabel 3.28 Atribut kelas DaftarPenjelasanActivity

Nama Atribut	*Visibility*	Tipe
menu	private	Menu
penjelasan1	private	ImageView
penjelasan2	private	ImageView
penjelasan3	private	ImageView
penjelasan4	private	ImageView

Adapun metode dari kelas DaftarPenjelasanActivity dapat dilihat pada Tabel 3.29.

Tabel 3.29 Metode kelas DaftarPenjelasanActivity

Nama Metode	*Visibility*	Tipe
onCreate(Bundle)	public	void

13. Kelas DaftarTesActivity

Kelas DaftarTesActivity merupakan kelas yang digunakan untuk mengontrol daftar tes daya ingat pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas DaftarTesActivity yang dapat dilihat pada Tabel 3.30.

Tabel 3.30 Atribut kelas DaftarTesActivity

Nama Atribut	*Visibility*	Tipe
daftar	private	Daftar
adapter	private	TesAdapter
helper	private	DBHelper

Adapun metode dari kelas DaftarTesActivity dapat dilihat pada Tabel 3.31.

Tabel 3.31 Metode kelas DaftarTesActivity

Nama Metode	*Visibility*	Tipe
onCreate(Bundle)	public	void
onDestroy()	public	void
onResume()	public	void
onListItemClick(ListView, View, int, long)	public	void

14. Kelas DBHelper

Kelas DBHelper merupakan kelas yang digunakan untuk mengontrol *database* pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas DBHelper yang dapat dilihat pada Tabel 3.32.

Tabel 3.32 Atribut kelas DBHelper

Nama Atribut	*Visibility*	Tipe
DATABASE_NAME	private	String

Nama Atribut	*Visibility*	Tipe
SCHEMA_VERSION	private	int
DATABASE_PATH	public	String
database	private	SQLiteOpenHelper
dbContext	private	Context

Adapun metode dari kelas DBHelper dapat dilihat pada Tabel 3.33.

Tabel 3.33 Metode kelas DBHelper

Nama Metode	*Visibility*	Tipe
DBHelper(Context)	public	
copyDataBase()	private	void
openDataBase()	public	void
checkDataBase()	private	void
onCreate(SQLiteDatabase)	public	void
onUpgrade(SQLiteDatabase, int, int)	public	void
getAll()	public	Cursor
getById(String)	public	Cursor
getAllTes()	public	Cursor
getByIdTes(String)	public	Cursor
getHasilTes()	public	Cursor
getKonten(String)	public	Cursor
updateJadwal(String, String)	public	void
updateTes(String, String)	public	void
insertHasilTes(String, String)	public	void
getData1(Cursor)	public	String
getData2(Cursor)	public	String
getData3(Cursor)	public	String
getData4(Cursor)	public	String

Nama Metode	*Visibility*	Tipe
getDatabase	public	SQLiteDatabase
setDatabase(SQLiteDatabase)	public	void

15. Kelas HasilTesActivity

Kelas HasilTesActivity merupakan kelas yang digunakan untuk mengontrol hasil tes pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas HasilTesActivity yang dapat dilihat pada Tabel 3.34.

Tabel 3.34 Atribut kelas HasilTesActivity

Nama Atribut	*Visibility*	Tipe
daftar	private	Daftar
adapter	private	HasilTesAdapter
helper	private	DBHelper

Adapun metode dari kelas HasilTesActivity dapat dilihat pada Tabel 3.35.

Tabel 3.35 Metode kelas HasilTesActivity

Nama Metode	*Visibility*	Tipe
onCreate(Bundle)	public	void
onDestroy()	public	void

16. Kelas HasilTesAdapter

Kelas HasilTesAdapter merupakan kelas yang digunakan untuk membangun koneksi dengan sumber data hasil tes pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini metode yang dimiliki oleh kelas HasilTesAdapter yang dapat dilihat pada Tabel 3.36.

Tabel 3.36 Metode kelas HasilTesAdapter

Nama Metode	*Visibility*	Tipe

Nama Metode	*Visibility*	Tipe
HasilTesAdapter(Cursor)	public	
bindView(View, Context, Cursor)	public	void
newView(Context, Cursor, ViewGroup)	public	View

17. Kelas HasilTesHolder

Kelas HasilTesHolder merupakan kelas yang digunakan untuk memetakan data hasil tes pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas HasilTesHolder yang dapat dilihat pada Tabel 3.37.

Tabel 3.37 Atribut kelas HasilTesHolder

Nama Atribut	*Visibility*	Tipe
bagian	private	TextView
skor	private	TextView

Adapun metode dari kelas HasilTesHolder dapat dilihat pada Tabel 3.38.

Tabel 3.38 Metode kelas HasilTesHolder

Nama Metode	*Visibility*	Tipe
HasilTesHolder(View, Typeface)	public	
PopulateForm(Cursor, DBHelper)	public	void

18. Kelas HitungSkor

Kelas HitungSkor merupakan kelas yang digunakan untuk menghitung skor latihan pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas HitungSkor yang dapat dilihat pada Tabel 3.39.

Tabel 3.39 Atribut kelas HitungSkor

Nama Atribut	*Visibility*	Tipe
skor	private	int
hasil	private	String

Adapun metode dari kelas HitungSkor dapat dilihat pada Tabel 3.40.

Tabel 3.40 Metode kelas HitungSkor

Nama Metode	*Visibility*	Tipe
getSkor()	protected	int
setSkor(int)	protected	void
getHasil()	protected	String
setHasil()	protected	void

19. Kelas JadwalActivity

Kelas JadwalActivity merupakan kelas yang digunakan untuk mengontrol jadwal latihan pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas JadwalActivity yang dapat dilihat pada Tabel 3.41.

Tabel 3.41 Atribut kelas JadwalActivity

Nama Atribut	*Visibility*	Tipe
jadwal	private	Daftar
adapter	private	JadwalAdapter
helper	private	DBHelper

Adapun metode dari kelas JadwalActivity dapat dilihat pada Tabel 3.42.

Tabel 3.42 Metode kelas JadwalActivity

Nama Metode	*Visibility*	Tipe
onCreate(Bundle)	public	void

Nama Metode	*Visibility*	Tipe
onDestroy()	public	void
onResume()	public	void
onListItemClick(ListView, View, int, long)	public	void

20. Kelas JadwalAdapter

Kelas JadwalAdapter merupakan kelas yang digunakan untuk membangun koneksi dengan sumber data jadwal latihan pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini metode yang dimiliki oleh kelas JadwalAdapter yang dapat dilihat pada Tabel 3.43.

Tabel 3.43 Metode kelas JadwalAdapter

Nama Metode	*Visibility*	Tipe
JadwalAdapter(Cursor)	public	
bindView(View, Context, Cursor)	public	void
newView(Context, Cursor, ViewGroup)	public	View

21. Kelas JadwalHolder

Kelas JadwalHolder merupakan kelas yang digunakan untuk memetakan data jadwal latihan pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas JadwalHolder yang dapat dilihat pada Tabel 3.44.

Tabel 3.44 Atribut kelas JadwalHolder

Nama Atribut	*Visibility*	Tipe
hari	private	TextView
materi	private	TextView
icon	private	ImageView

Adapun metode dari kelas JadwalHolder dapat dilihat pada Tabel 3.45.

Tabel 3.45 Metode kelas JadwalHolder

Nama Metode	*Visibility*	Tipe
JadwalHolder(View, Typeface)	public	
PopulateForm(Cursor, JadwalHelper)	public	void

22. Kelas LatihanActivity

Kelas LatihanActivity merupakan kelas yang digunakan untuk mengontrol latihan n-back pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas LatihanActivity yang dapat dilihat pada Tabel 3.46.

Tabel 3.46 Atribut kelas LatihanActivity

Nama Atribut	*Visibility*	Tipe
latihan	private	Latihan
hitung	private	HitungSkor
handler	private	Handler
randomGenerator	private	Random
helper	private	DBHelper
konten	private	TextView
hari	private	TextView
materi	private	TextView
tutorial	private	TextView
mulai	private	ImageView
sama	private	ImageView
previous	private	ImageButton
next	private	ImageButton
is	private	ImageSwitcher

Nama Atribut	*Visibility*	Tipe
layout	private	LinearLayout
angka_acak	private	int[]
imageIds	private	int[]
i	private	int
j	private	int
nCounter	private	int
runnable	private	Runnable

Adapun metode dari kelas LatihanActivity dapat dilihat pada Tabel 3.47.

Tabel 3.47 Metode kelas LatihanActivity

Nama Metode	*Visibility*	Tipe
onCreate(Bundle)	public	void
onDestroy()	public	void
tampilHasil()	private	void
tampilkanKonten()	private	void
setKonten()	private	void
acakAngka()	private	void
isiStimulus()	private	void
setTutorial()	private	void

23. Kelas MainActivity

Kelas MainActivity merupakan kelas yang digunakan untuk mengontrol menu utama pada aplikasi *mobile* “Latihan Daya Ingat”. Berikut ini atribut yang dimiliki oleh kelas MainActivity yang dapat dilihat pada Tabel 3.48.

Tabel 3.48 Atribut kelas MainActivity

Nama Atribut	*Visibility*	Tipe
menu	private	Menu
menu1	private	ImageView
menu2	private	ImageView
menu3	private	ImageView
menu4	private	ImageView

Adapun metode dari kelas MainActivity dapat dilihat pada Tabel 3.49.

Tabel 3.49 Metode kelas MainActivity

Nama Metode	*Visibility*	Tipe
onCreate(Bundle)	public	void

24. Kelas PenjelasanActivity

Kelas PenjelasanActivity merupakan kelas yang digunakan untuk mengontrol penjelasan pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas PenjelasanActivity yang dapat dilihat pada Tabel 3.50.

Tabel 3.50 Atribut kelas PenjelasanActivity

Nama Atribut	*Visibility*	Tipe
penjelasan	private	Penjelasan
i	private	int

Adapun metode dari kelas PenjelasanActivity dapat dilihat pada Tabel 3.51.

Tabel 3.51 Metode kelas PenjelasanActivity

Nama Metode	*Visibility*	Tipe
onCreate(Bundle)	public	void
setTutorial()	private	void

25. Kelas TesActivity

Kelas TesActivity merupakan kelas yang digunakan untuk mengontrol tes daya ingat pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas TesActivity yang dapat dilihat pada Tabel 3.52.

Tabel 3.52 Atribut kelas TesActivity

Nama Atribut	*Visibility*	Tipe
tes	private	Tes
handler	private	Handler
helper	private	DBHelper
konten	private	TextView
no_tes	private	TextView
bagian	private	TextView
tutorial	private	TextView
mulai	private	ImageView
simpan	private	ImageView
previous	private	ImageButton
next	private	ImageButton
is	private	ImageSwitcher
layout	private	LinearLayout
form	private	LinearLayout
entry1	private	EditText
entry2	private	EditText
entry3	private	EditText
entry4	private	EditText
entry5	private	EditText
entry6	private	EditText
imageIds	private	int[]
kunci	private	String[]
kata_split	private	String[]

Nama Atribut	*Visibility*	Tipe
i	private	int
idx	private	int
ik	private	int
j	private	int
jml_benar	private	int
delay	private	int
nCounter	private	int
marquee	private	String
runnable	private	Runnable

Adapun metode dari TesActivity dapat dilihat pada Tabel 3.53.

Tabel 3.53 Metode kelas TesActivity

Nama Metode	*Visibility*	Tipe
onCreate(Bundle)	public	void
onDestroy()	public	void
tampilkanKonten()	private	void
start()	private	void
stop()	private	void
setKonten()	private	void
setTutorial(int)	private	void

26. Kelas TesAdapter

Kelas TesAdapter merupakan kelas yang digunakan untuk membangun koneksi dengan sumber data daftar tes daya ingat pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini metode yang dimiliki oleh kelas TesAdapter yang dapat dilihat pada Tabel 3.54.

Tabel 3.54 Metode kelas TesAdapter

Nama Metode	*Visibility*	Tipe
TesAdapter(Cursor)	public	
bindView(View, Context, Cursor)	public	void
newView(Context, Cursor, ViewGroup)	public	View

27. Kelas TesHolder

Kelas TesHolder merupakan kelas yang digunakan untuk memetakan data daftar tes pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas TesHolder yang dapat dilihat pada Tabel 3.55.

Tabel 3.55 Atribut kelas TesHolder

Nama Atribut	*Visibility*	Tipe
id	private	TextView
status	private	TextView
icon	private	ImageView

Adapun metode dari kelas TesHolder dapat dilihat pada Tabel 3.56.

Tabel 3.56 Metode kelas TesHolder

Nama Metode	*Visibility*	Tipe
TesHolder(View, Typeface)	public	
PopulateForm(Cursor, JadwalHelper)	public	void

28. Kelas Daftar

Kelas Daftar merupakan kelas yang digunakan untuk menampung data daftar *item* pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas Daftar yang dapat dilihat pada Tabel 3.57.

Tabel 3.57 Atribut kelas Daftar

Nama Atribut	*Visibility*	Tipe
model	private	Cursor

Adapun metode dari kelas Daftar dapat dilihat pada Tabel 3.58.

Tabel 3.58 Metode kelas Daftar

Nama Metode	*Visibility*	Tipe
getModel()	protected	Cursor
setModel(Cursor)	protected	void

29. Kelas Latihan

Kelas Latihan merupakan kelas yang digunakan untuk menampung data latihan pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas Latihan yang dapat dilihat pada Tabel 3.59.

Tabel 3.59 Atribut kelas Latihan

Nama Atribut	*Visibility*	Tipe
waktu	private	int
nilai_n	private	int
id_extra	private	int
id_extra_str	private	String
idplus_str	private	String
status	private	String
materi	private	String

Nama Atribut	*Visibility*	Tipe
stimuli	private	String
stimulus	private	String[]
final_stimulus	private	String[]

Adapun metode dari kelas Latihan dapat dilihat pada Tabel 3.60.

Tabel 3.60 Metode kelas Latihan

Nama Metode	*Visibility*	Tipe
getWaktu()	protected	int
setWaktu(int)	protected	void
getNilai_n()	protected	int
setNilai_n(int)	protected	void
getId_extra()	protected	int
setId_extra(int)	protected	void
getId_extra_str()	protected	String
setId_extra_str(String)	protected	void
getIdplus_str()	protected	String
setIdplus_str(String)	protected	void
getStatus()	protected	String
setStatus(String)	protected	void
getMateri()	protected	String
setMateri(String)	protected	void
getStimuli()	protected	String
setStimuli(String)	protected	void
getStimulus()	protected	String[]
setStimulus(String[])	protected	void
getFinal_stimulus()	protected	String[]
setFinal_stimulus(int, String)	protected	void
setLengthFinal_stimulus(String[])	protected	void

30. Kelas Menu

Kelas Menu merupakan kelas yang digunakan untuk menampung data menu pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas Menu yang dapat dilihat pada Tabel 3.61.

Tabel 3.61 Atribut kelas Menu

Nama Atribut	*Visibility*	Tipe
intent	private	Intent

Adapun metode dari kelas Menu dapat dilihat pada Tabel 3.62.

Tabel 3.62 Metode kelas Menu

Nama Metode	*Visibility*	Tipe
getIntent()	protected	Intent
setIntent(Intent)	protected	void

31. Kelas Penjelasan

Kelas Penjelasan merupakan kelas yang digunakan untuk menampung data penjelasan pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas Penjelasan yang dapat dilihat pada Tabel 3.63.

Tabel 3.63 Atribut kelas Penjelasan

Nama Atribut	*Visibility*	Tipe
value	private	int
imageIds	private	int[]

Adapun metode dari kelas Penjelasan dapat dilihat pada Tabel 3.64.

Tabel 3.64 Metode kelas Penjelasan

Nama Metode	*Visibility*	Tipe
getValue()	protected	int
setValue(int)	protected	void
getImageIds()	protected	int[]
setImageIds(int[])	protected	void

32. Kelas Tes

Kelas Tes merupakan kelas yang digunakan untuk menampung data tes pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini atribut yang dimiliki oleh kelas Tes yang dapat dilihat pada Tabel 3.65.

Tabel 3.65 Atribut kelas Tes

Nama Atribut	*Visibility*	Tipe
kalimat	private	String[]
span	private	int
id_extra	private	String
skor	private	int

Adapun metode dari kelas Tes dapat dilihat pada Tabel 3.66.

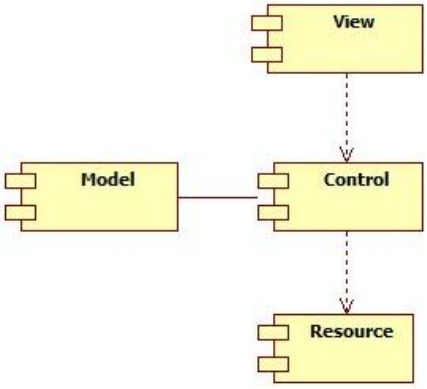
Tabel 3.66 Metode kelas Tes

Nama Metode	*Visibility*	Tipe
getKalimat()	protected	String[]
setKalimat(String[])	protected	void
getSpan()	protected	int
setSpan(int)	protected	void
getId_extra()	protected	String

Nama Metode	*Visibility*	Tipe
setId_extra (String)	protected	void
getSkor()	protected	int
setSkor(int, int)	protected	void

3.1.6.5 *Component Diagram*

Component diagram digunakan untuk memecah suatu sistem menjadi beberapa bagian komponen dan menampilkan hubungan mereka.



Gambar 3.26 *Component Diagram* Aplikasi "Latihan Daya Ingat"

3.1.6.6 *Deployment Diagram*

Deployment diagram menggambarkan bagaimana infrastruktur sistem dari aplikasi *mobile* "Latihan Daya Ingat". Adapun *deployment diagram* aplikasi *mobile* "Latihan Daya Ingat" dapat dilihat pada Gambar 3.27.



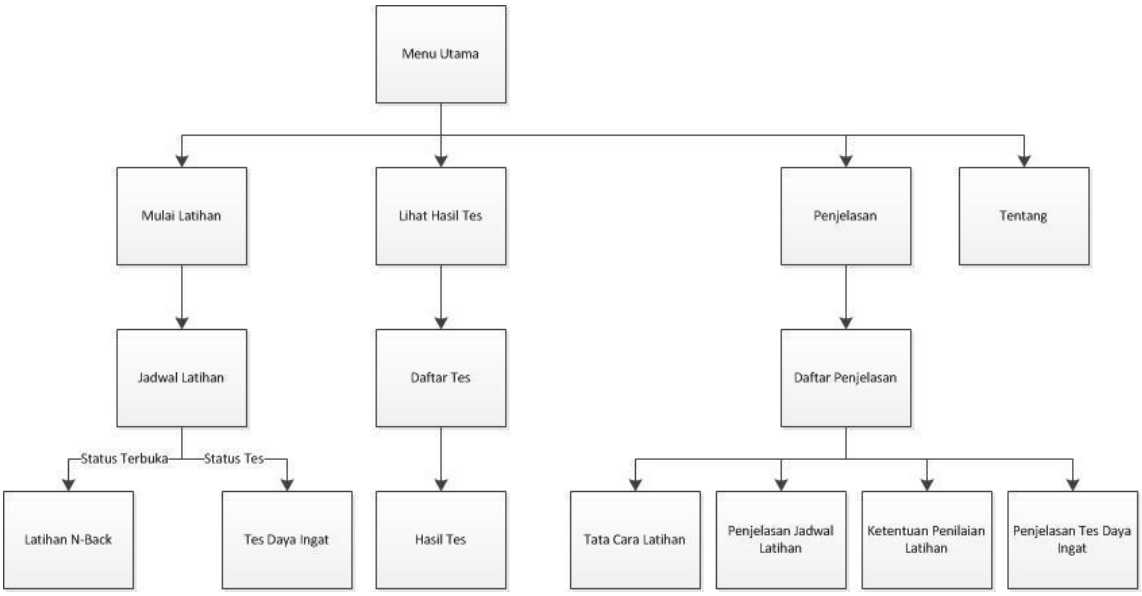
Gambar 3.27 *Deployment Diagram* Aplikasi "Latihan Daya Ingat"

3.2 Perancangan Sistem

Perancangan sistem merupakan penggambaran, perencanaan, dan pembuatan dari beberapa elemen sistem yang terpisah ke dalam suatu kesatuan model yang utuh berdasarkan analisis sistem yang telah dilakukan sebelumnya. Tahapan perancangan pembangunan aplikasi *mobile* "Latihan Daya Ingat" ini meliputi perancangan struktur menu, perancangan antarmuka, dan jaringan semantik.

3.2.1 Perancangan Struktur Menu

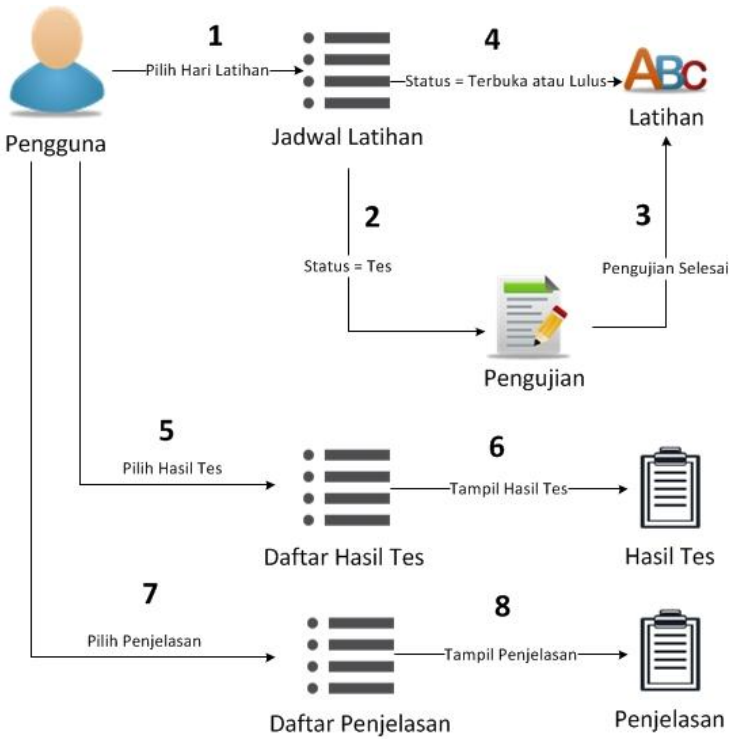
Perancangan struktur menu merupakan gambaran jalur pemakaian aplikasi sehingga aplikasi yang dibangun mudah dipahami dan mudah digunakan. Perancangan struktur menu dari aplikasi *mobile* "Latihan Daya Ingat" dapat dilihat pada Gambar 3.28.



Gambar 3.28 Struktur Menu Aplikasi "Latihan Daya Ingat"

3.2.2 Perancangan Kerangka Kerja

Perancangan kerangka kerja menggambarkan alur dari penggunaan aplikasi “Latihan Daya Ingat”. Adapun alur penggunaan aplikasi “Latihan Daya Ingat” dapat dilihat pada Gambar 3.29.



Gambar 3.29 Perancangan Kerangka Kerja

Berikut adalah keterangan mengenai perancangan kerja pada Gambar 3.34:

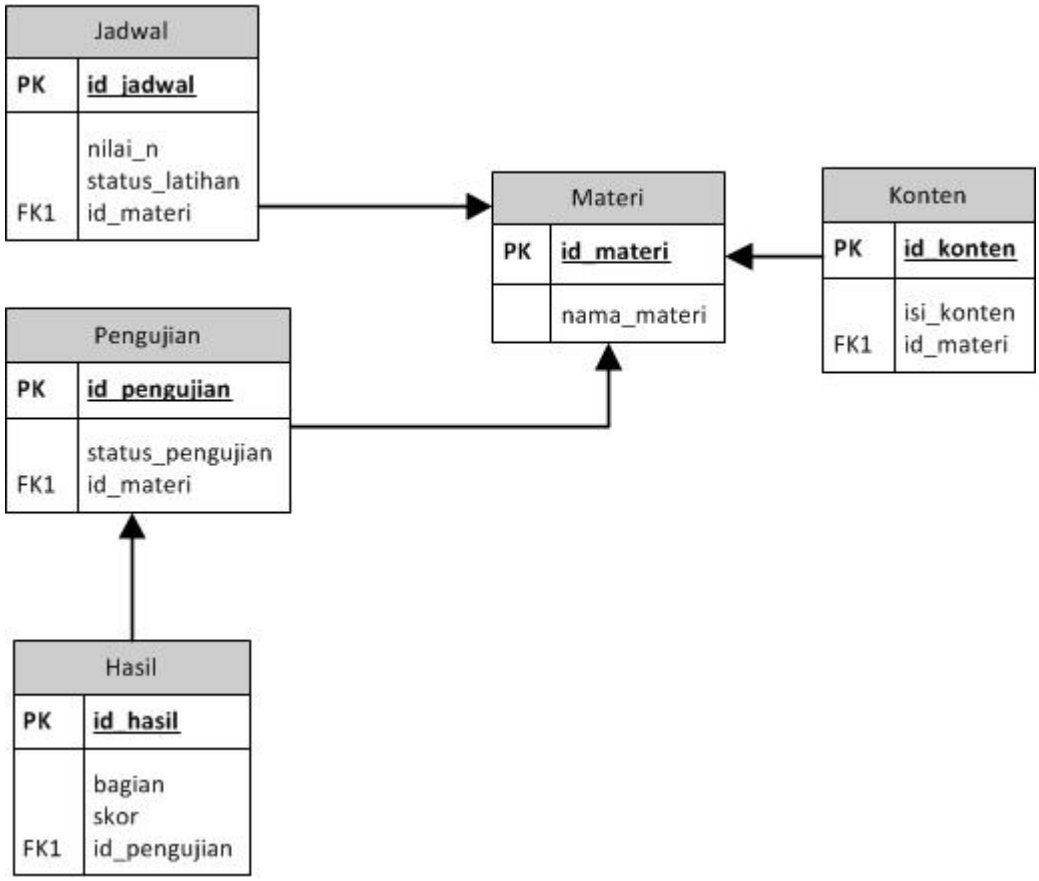
1. Pengguna memilih hari latihan.
2. Status latihan yang dipilih adalah Tes, maka sistem akan menampilkan pengujian kemampuan daya ingat.
3. Setelah pengujian selesai, sistem menampilkan latihan.
4. Status latihan yang dipilih adalah Terbuka atau Lulus, maka sistem akan menampilkan latihan daya ingat.
5. Pengguna memilih hasil tes yang akan ditampilkan.
6. Sistem menampilkan hasil tes yang dipilih.
7. Pengguna memilih pengujian yang akan ditampilkan.
8. Sistem menampilkan pengujian yang dipilih.

3.2.3 Perancangan Data

Perancangan data atau lebih dikenal dengan perancangan basis data yaitu menciptakan atau merancang data yang terhubung dan disimpan secara bersama-sama. Untuk menggambarkannya digunakanlah skema relasi dan perancangan struktur tabel. Dari dua hasil tersebut, implementasi basis data akan dapat dikerjakan.

3.2.3.1 Skema Relasi

Setiap tabel haruslah memiliki hubungan dengan tabel yang lainnya. Bila tidak ada penghubungan antar tabel maka dapat dikatakan pemodelan untuk membuat satu basis data adalah gagal. Adapun gambaran skema relasi yang ada dapat dilihat pada Gambar 3.30.



Gambar 3.30 Skema Relasi

3.2.3.2 Struktur Tabel

Berikut ini adalah tabel-tabel yang terdapat pada basis data yang digunakan dalam pembangunan aplikasi "Latihan Daya Ingat":

1. Tabel Materi

Tabel 3.67 Struktur Tabel Materi

Nama Field	Tipe Data	Panjang	Atribut	Keterangan
id_materi	Integer	-	*Primary key*	*Auto_increment*
nama_materi	Varchar	15		*Not Null*

2. Tabel Konten

Tabel 3.68 Struktur Tabel Konten

Nama Field	Tipe Data	Panjang	Atribut	Keterangan
id_konten	Integer	-	*Primary key*	*Auto_increment*
isi_konten	Text	-		*Not Null*
id_materi	Integer	-	*Foreign key* (Materi.id_materi)	*Not Null*

3. Tabel Jadwal

Tabel 3.69 Struktur Tabel Jadwal

Nama Field	Tipe Data	Panjang	Atribut	Keterangan
id_jadwal	Integer	-	*Primary key*	*Auto_increment*
nilai_n	Integer	-		*Not Null*
status_latihan	Varchar	15		*Not Null*
id_materi	Integer	-	*Foreign key* (Materi.id_materi)	*Not Null*

4. Tabel Pengujian

Tabel 3.70 Struktur Tabel Pengujian

Nama Field	Tipe Data	Panjang	Atribut	Keterangan
id_pengujian	Integer	-	*Primary key*	*Auto_increment*
status_pengujian	Varchar	15		*Not Null*
id_materi	Integer	-	*Foreign key* (Materi.id_materi)	*Not Null*

5. Tabel Hasil

Tabel 3.71 Struktur Tabel Hasil

Nama Field	Tipe Data	Panjang	Atribut	Keterangan
id_hasil	Integer	-	*Primary key*	*Auto_increment*
bagian	Integer	-		*Not Null*
skor	Varchar	5		*Not Null*
id_pengujian	Integer	-	*Foreign key* (Pengujain.id_pengujian)	*Not Null*

3.2.4 Perancangan Antarmuka

Perancangan antarmuka bertujuan untuk memberikan gambaran tentang aplikasi yang akan dibangun, sehingga akan mempermudah dalam membangun aplikasi. Perancangan antarmuka yang akan diterapkan pada aplikasi nantinya terdiri dari perancangan *form* dan juga perancangan pesan.

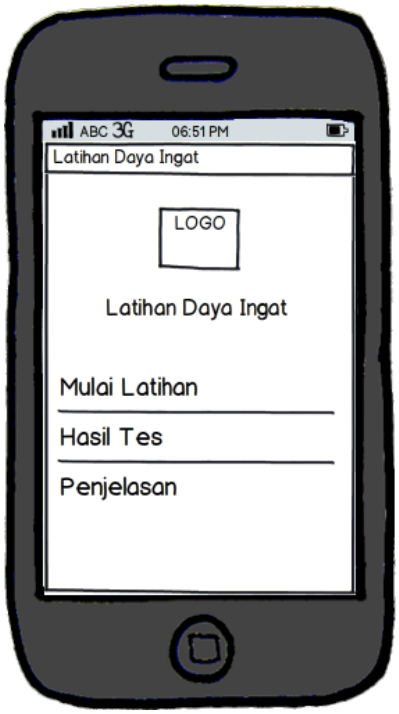
3.2.4.1 Perancangan *Form*

Perancangan *form* mendeskripsikan rencana tampilan dari setiap *form* yang akan digunakan pada tampilan aplikasi *mobile* "Latihan Daya Ingat". Berikut ini gambaran dari perancangan *form* aplikasi *mobile* "Latihan Daya Ingat" seperti terlihat pada Gambar 3.31 sampai dengan Gambar 3.41.

1. Perancangan *Form* Menu Utama

Perancangan *form* menu utama merupakan rancangan menu yang akan muncul pertama kali pada saat pengguna membuka aplikasi. Adapun perancangan *form* menu utama dapat dilihat pada Gambar 3.31.

A01



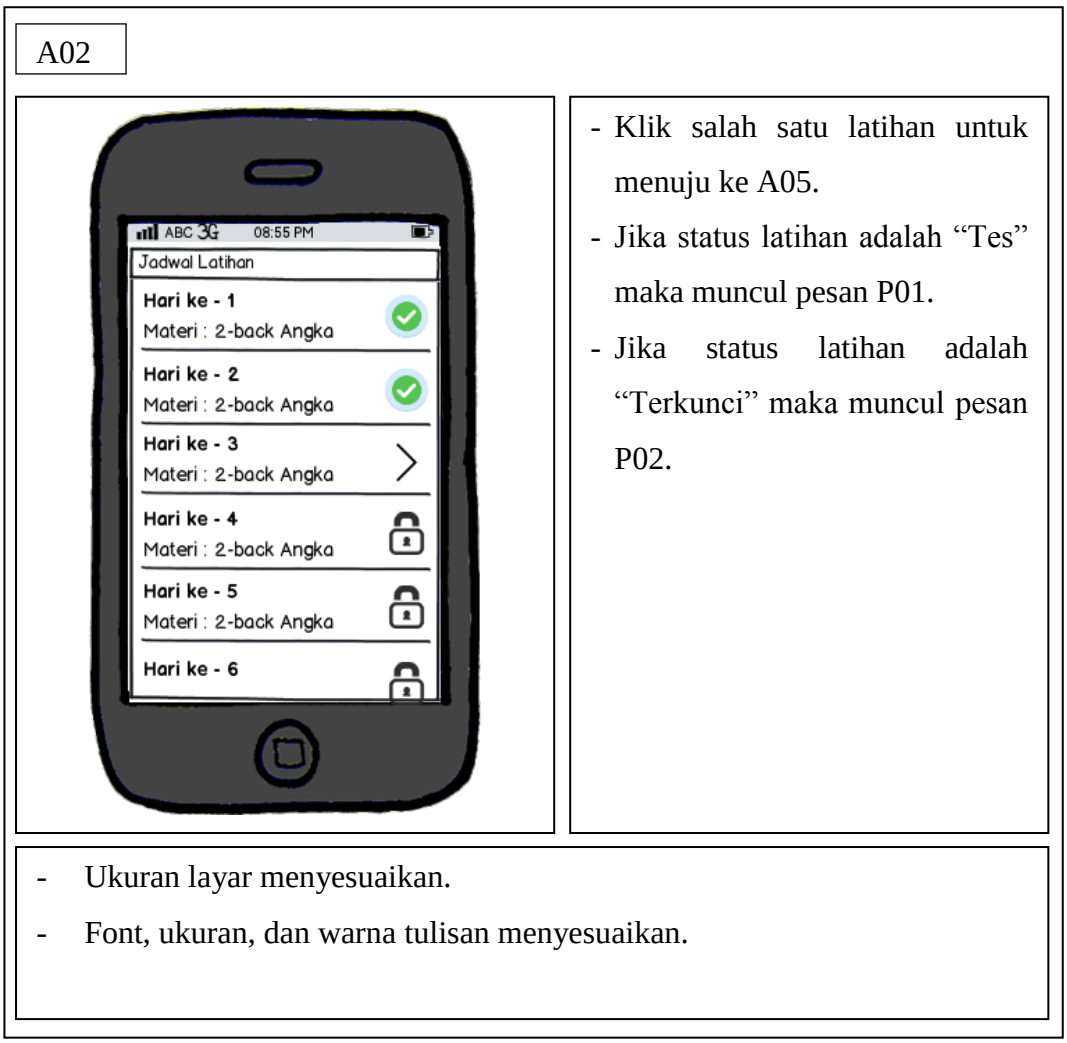
- Klik menu "Mulai Latihan" untuk menuju ke A02.
- Klik menu "Hasil Tes" untuk menuju ke A03.
- Klik menu "Penjelasan" untuk menuju ke A04.

- Ukuran layar menyesuaikan.
- Font, ukuran, dan warna tulisan menyesuaikan.

Gambar 3.31 Perancangan *Form* Menu Utama

2. Perancangan *Form* Jadwal Latihan

Perancangan *form* jadwal latihan merupakan rancangan menu yang akan menampilkan jadwal latihan n-back. *Form* jadwal latihan akan muncul setelah pengguna memilih menu "Mulai Latihan". Adapun perancangan *form* jadwal latihan dapat dilihat pada Gambar 3.32.

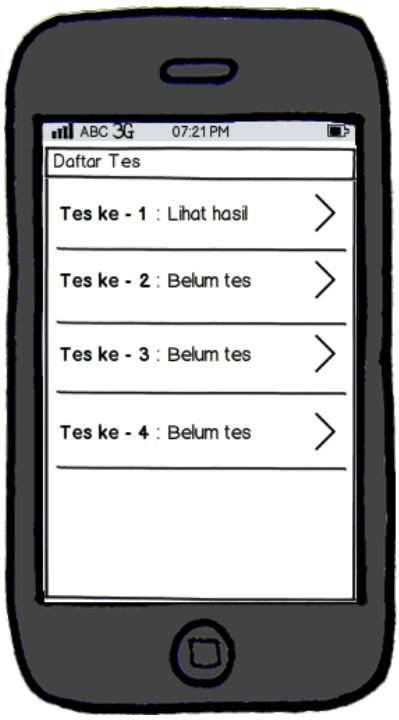


Gambar 3.32 Perancangan *Form* Jadwal Latihan

3. Perancangan *Form* Daftar Tes

Perancangan *form* daftar tes merupakan rancangan menu yang akan menampilkan daftar tes daya ingat. *Form* daftar tes akan muncul setelah pengguna memilih menu “Hasil Tes”. Adapun perancangan *form* daftar tes dapat dilihat pada Gambar 3.33.

A03



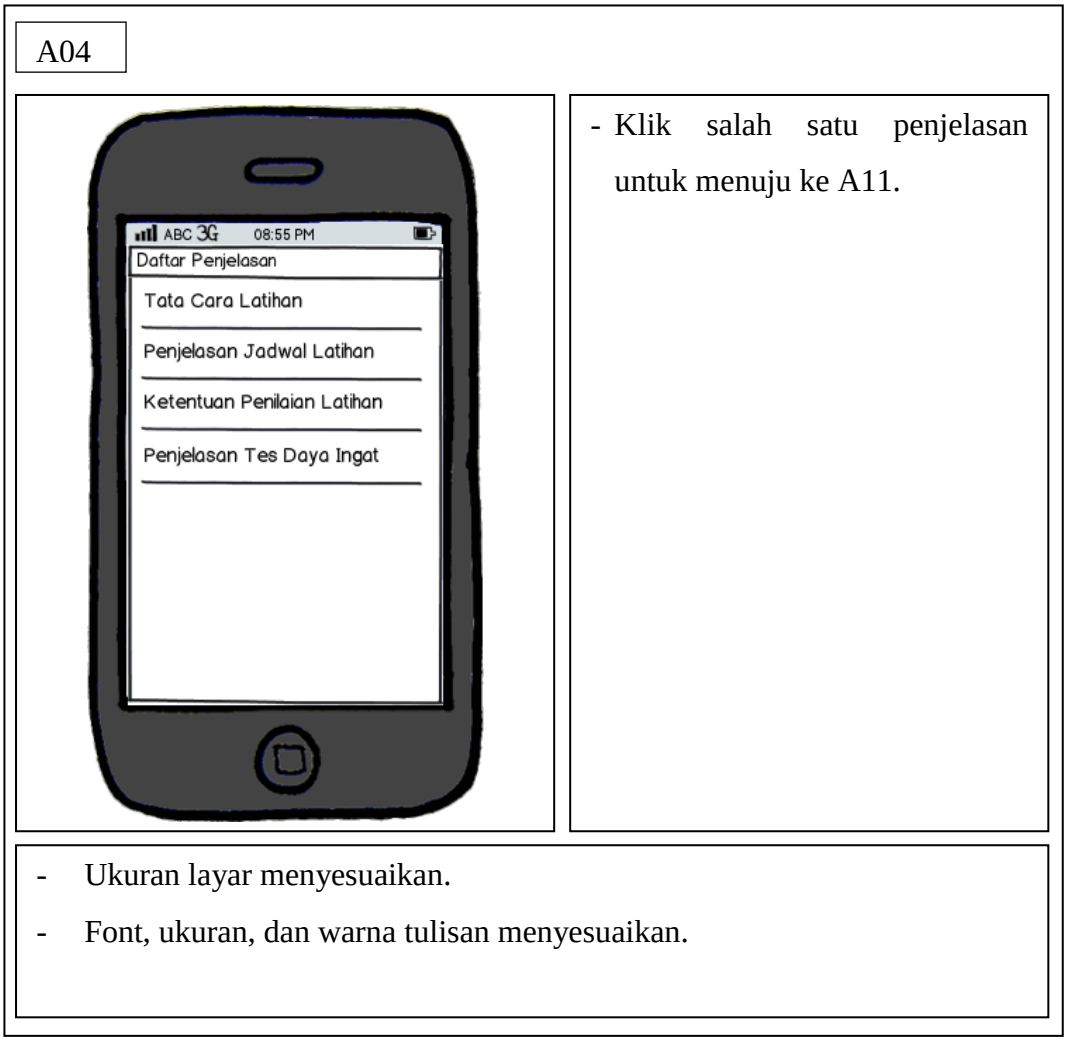
- Klik salah satu tes untuk menuju ke A10.
- Jika belum tes dan diklik maka muncul pesan P03.

- Ukuran layar menyesuaikan.
- Font, ukuran, dan warna tulisan menyesuaikan.

Gambar 3.33 Perancangan *Form* Jadwal Tes

4. Perancangan *Form* Daftar Penjelasan

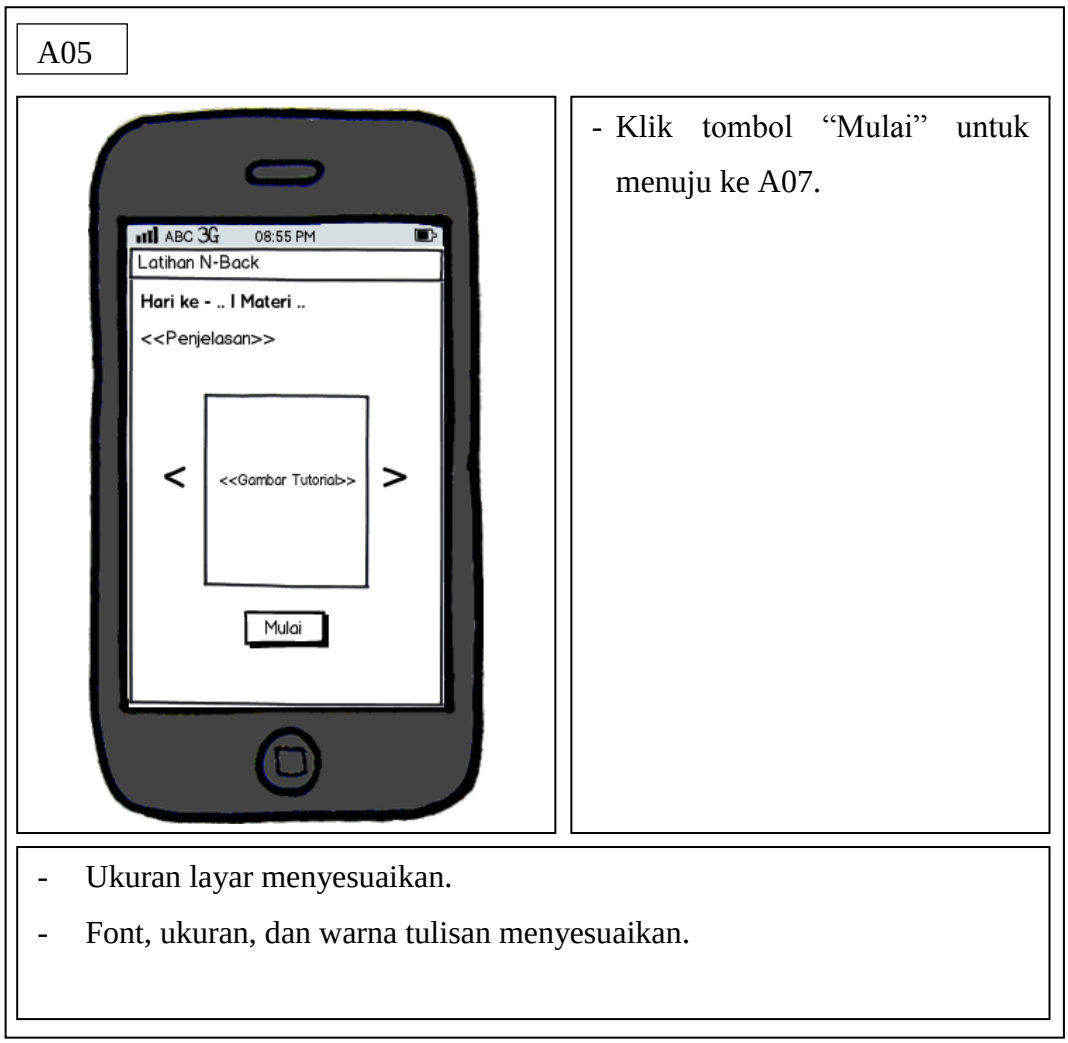
Perancangan *form* daftar penjelasan merupakan rancangan menu yang akan menampilkan daftar penjelasan mengenai fitur – fitur yang terdapat pada aplikasi. *Form* daftar penjelasan akan muncul setelah pengguna memilih menu "Penjelasan". Adapun perancangan *form* daftar penjelasan dapat dilihat pada Gambar 3.34.



Gambar 3.34 Perancangan *Form* Daftar Penjelasan

5. Perancangan *Form Tutorial* Latihan N-Back

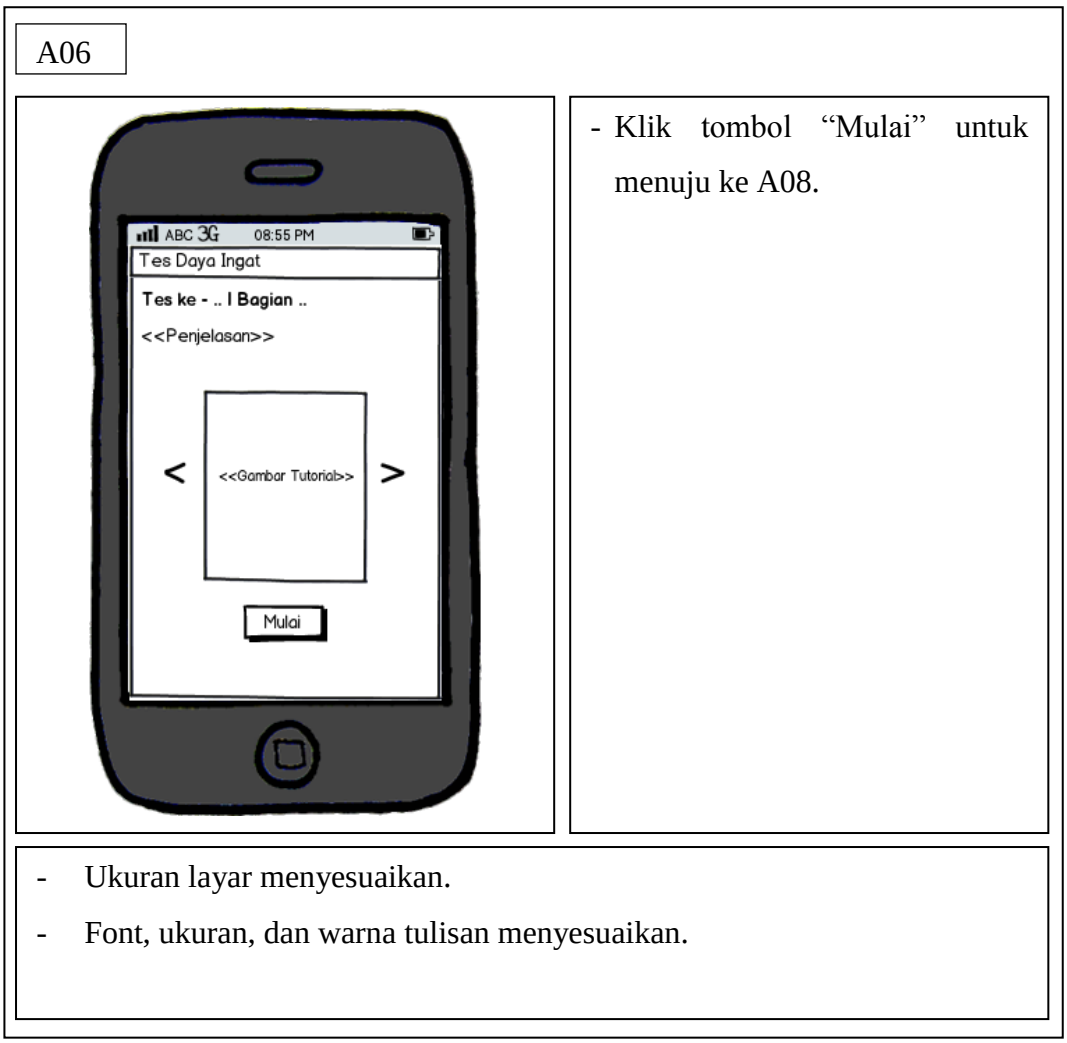
Perancangan *form tutorial* latihan n-back merupakan rancangan antarmuka yang akan menampilkan *tutorial* latihan n-back. *Form tutorial* latihan n-back akan muncul setelah pengguna memilih salah satu hari pada jadwal latihan. Adapun perancangan *form tutorial* latihan n-back dapat dilihat pada Gambar 3.35.



Gambar 3.35 Perancangan *Form Tutorial* Latihan N-Back

6. Perancangan *Form Tutorial* Tes Daya Ingat

Perancangan *form tutorial* tes daya ingat merupakan rancangan antarmuka yang akan menampilkan *tutorial* tes daya ingat. *Form tutorial* tes daya ingat akan muncul setelah pengguna memilih salah satu tes pada jadwal tes. Adapun perancangan *form tutorial* tes daya ingat dapat dilihat pada Gambar 3.36.



Gambar 3.36 Perancangan *Form Tutorial* Tes Daya Ingat

7. Perancangan *Form* Latihan N-Back

Perancangan *form* latihan n-back merupakan rancangan antarmuka yang akan menampilkan latihan n-back. *Form* latihan n-back akan muncul setelah pengguna klik tombol "Mulai" pada *form tutorial* latihan n-back. Adapun perancangan *form* latihan n-back dapat dilihat pada Gambar 3.37.

A07



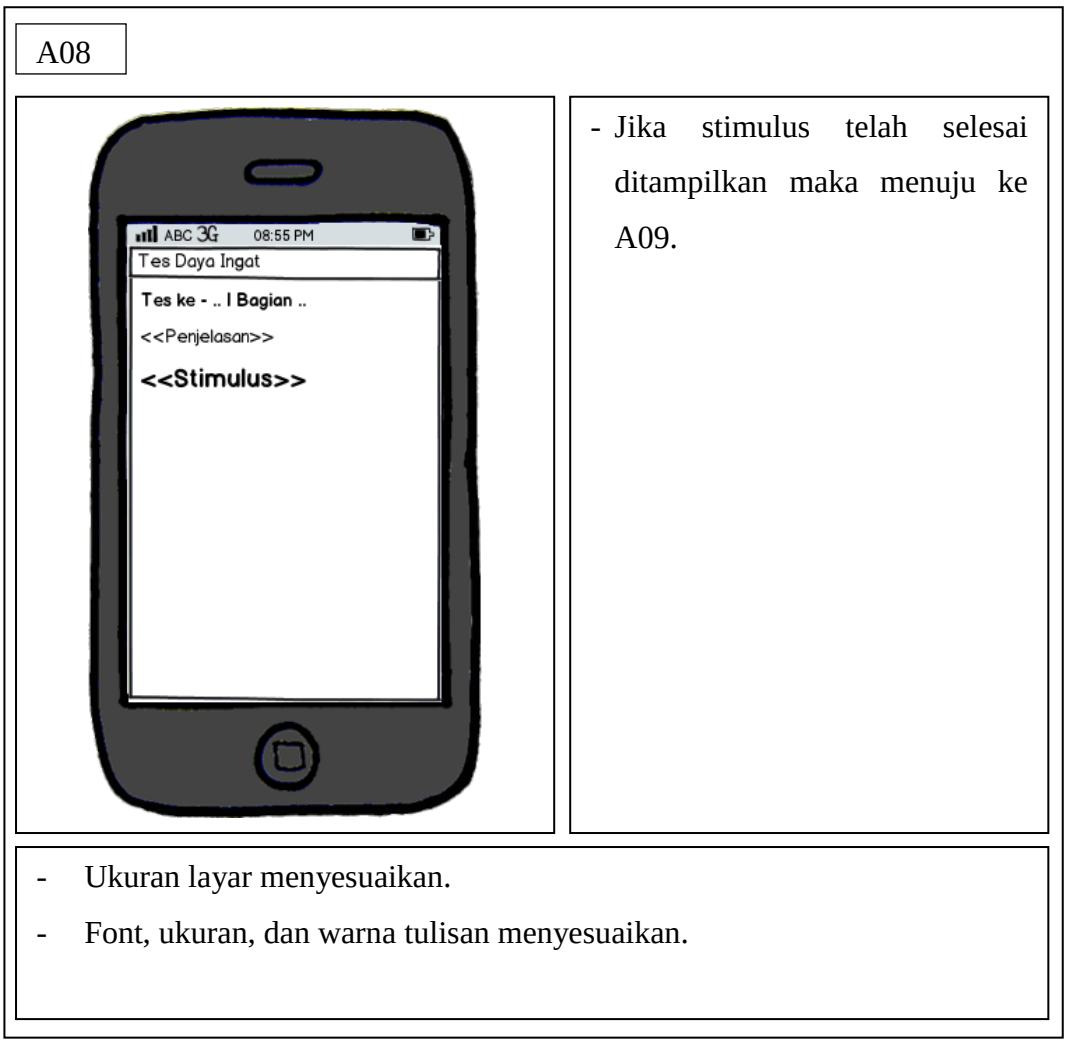
- Klik tombol "Sama" untuk menjawab.
- Jika latihan telah selesai dan lulus muncul pesan P04.
- Jika latihan telah selesai dan tidak lulus muncul pesan P05.

- Ukuran layar menyesuaikan.
- Font, ukuran, dan warna tulisan menyesuaikan.

Gambar 3.37 Perancangan *Form* Latihan N-Back

8. Perancangan *Form* Tes Daya Ingat

Perancangan *form* tes daya ingat merupakan rancangan antarmuka yang akan menampilkan tes daya ingat. *Form* tes daya ingat akan muncul setelah pengguna klik tombol "Mulai" pada *form tutorial* tes daya ingat. Adapun perancangan *form* tes daya ingat dapat dilihat pada Gambar 3.38.



Gambar 3.38 Perancangan *Form* Tes Daya Ingat

9. Perancangan *Form Input* Jawaban Tes Daya Ingat

Perancangan *form input* jawaban tes daya ingat merupakan rancangan antarmuka yang akan menampilkan *form input* jawaban tes daya ingat. *Form input* jawaban tes daya ingat digunakan untuk memasukkan jawaban pada tes. Adapun perancangan *form input* jawaban tes daya ingat dapat dilihat pada Gambar 3.39.

A09



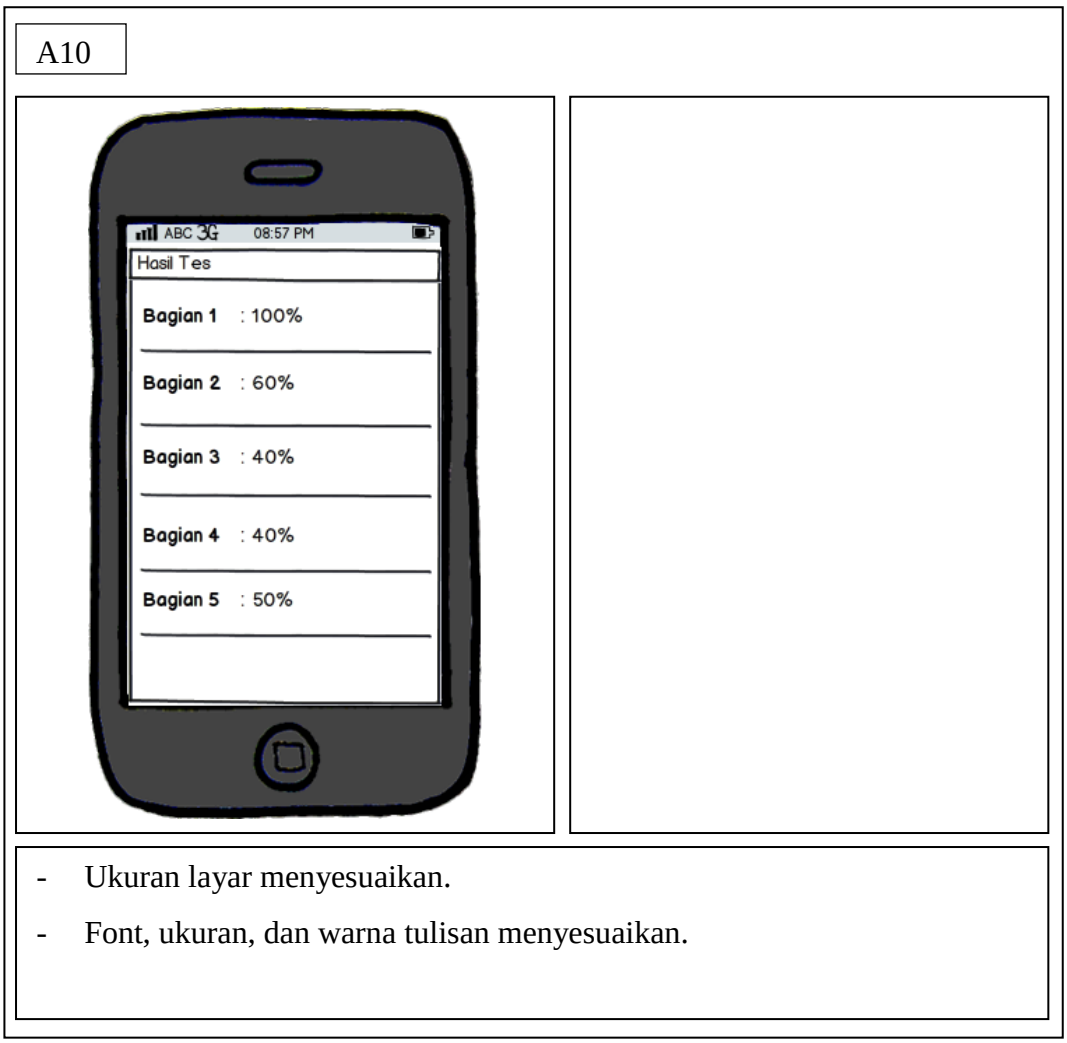
- Klik tombol "Simpan" untuk menyimpan jawaban.
- Jika bagian tes belum habis maka menuju ke A08.
- Jika bagian tes telah habis maka menuju ke A06.
- Jika tes telah selesai maka menuju ke A02.

- Ukuran layar menyesuaikan.
- Font, ukuran, dan warna tulisan menyesuaikan.

Gambar 3.39 Perancangan *Form Input* Jawaban Tes Daya Ingat

10. Perancangan *Form* Hasil Tes

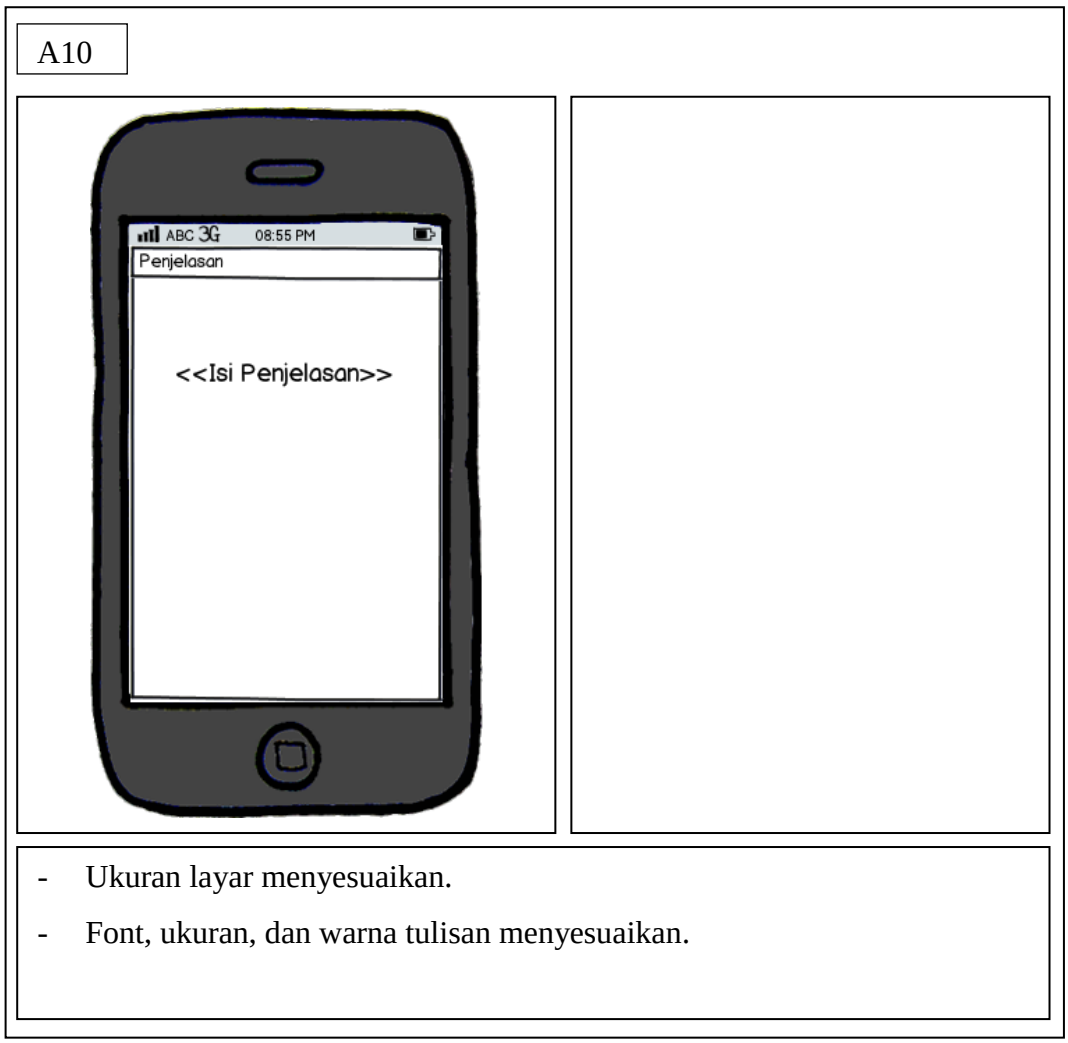
Perancangan *form* hasil tes merupakan rancangan antarmuka yang akan menampilkan hasil tes pada tiap bagian tes. *Form* hasil tes akan muncul setelah pengguna memilih salah satu tes pada daftar tes. Adapun perancangan *form* hasil tes dapat dilihat pada Gambar 3.40.



Gambar 3.40 Perancangan *Form* Hasil Tes

11. Perancangan *Form* Penjelasan

Perancangan *form* penjelasan merupakan rancangan antarmuka yang akan menampilkan penjelasan mengenai fitur pada aplikasi. *Form* penjelasan akan muncul setelah pengguna memilih salah satu penjelasan pada daftar penjelasan. Adapun perancangan *form* penjelasan dapat dilihat pada Gambar 3.41.



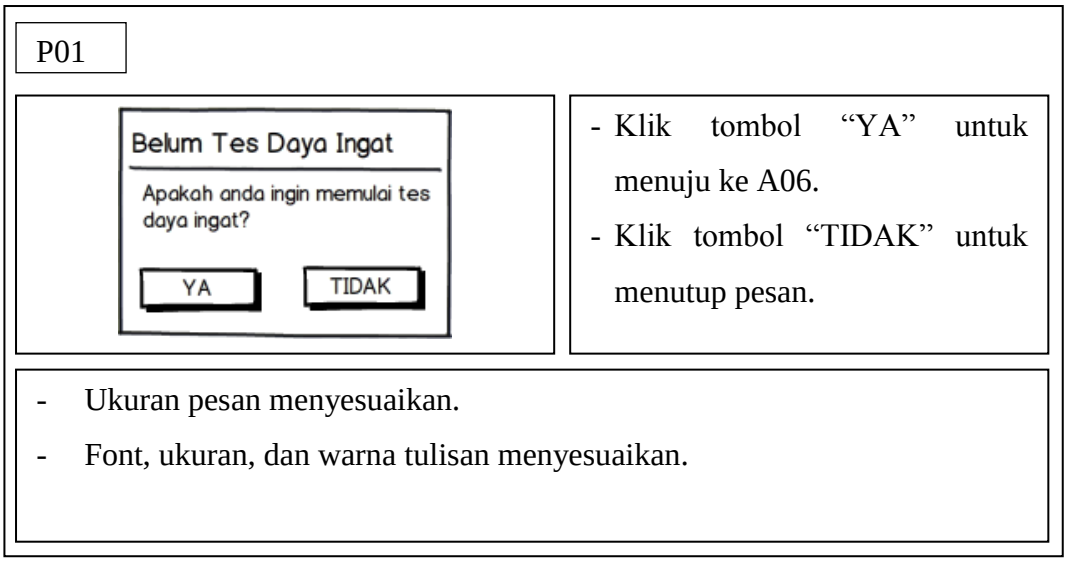
Gambar 3.41 Perancangan *Form* Penjelasan

3.2.4.2 Perancangan Pesan

Perancangan pesan mendeskripsikan rencana tampilan dari setiap pesan yang digunakan pada aplikasi *mobile* "Latihan Daya Ingat". Berikut ini gambaran dari perancangan pesan pada aplikasi *mobile* "Latihan Daya Ingat" seperti terlihat pada Gambar 3.42 sampai dengan Gambar 3.46.

1. Perancangan Pesan Belum Tes

Pesan belum tes berisi informasi bahwa pengguna belum melakukan tes daya ingat. Pesan tersebut muncul pada saat pengguna memilih latihan yang terdapat ikon *warning*. Adapun gambaran umum dari perancangan pesan belum tes pada aplikasi *mobile* "Latihan Daya Ingat" dapat dilihat pada Gambar 3.42.

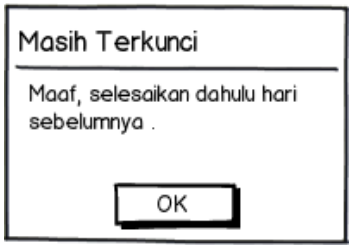


Gambar 3.42 Perancangan Pesan Belum Tes

2. Perancangan Pesan Masih Terkunci

Pesan masih terkunci berisi informasi bahwa latihan tidak dapat dipilih karena pengguna belum lulus latihan sebelumnya. Pesan tersebut muncul pada saat pengguna memilih latihan yang terdapat ikon gembok. Adapun gambaran umum dari perancangan pesan masih terkunci pada aplikasi *mobile* "Latihan Daya Ingat" dapat dilihat pada Gambar 3.43.

P02



- Klik tombol "OK" untuk menutup pesan.

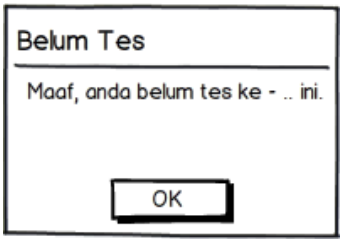
- Ukuran pesan menyesuaikan.
- Font, ukuran, dan warna tulisan menyesuaikan.

Gambar 3.43 Perancangan Pesan Masih Terkunci

3. Perancangan Pesan Belum Tes

Pesan belum tes berisi informasi bahwa pengguna belum melakukan tes daya ingat. Pesan tersebut muncul pada saat pengguna memilih salah satu tes dan belum melakukan tes yang dipilih sehingga tidak ada hasil tes pada tes tersebut. Adapun gambaran umum dari perancangan pesan belum tes pada aplikasi *mobile* "Latihan Daya Ingat" dapat dilihat pada Gambar 3.44.

P03



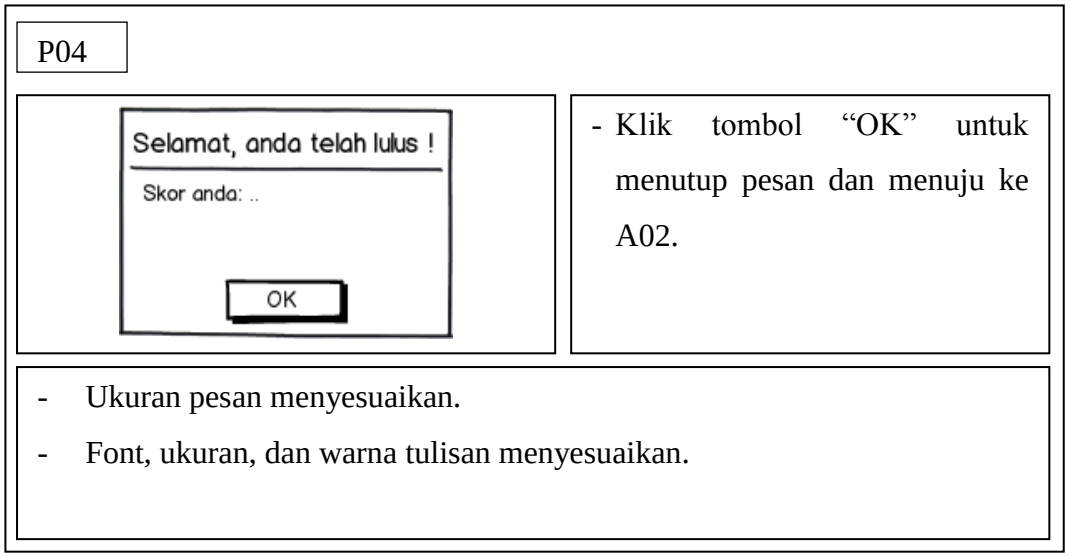
- Klik tombol "OK" untuk menutup pesan.

- Ukuran pesan menyesuaikan.
- Font, ukuran, dan warna tulisan menyesuaikan.

Gambar 3.44 Perancangan Pesan Belum Tes

4. Perancangan Pesan Telah Lulus Latihan

Pesan telah lulus latihan berisi informasi bahwa pengguna telah lulus latihan beserta skor yang diperoleh. Pesan tersebut muncul pada saat pengguna telah menyelesaikan latihan dan lulus. Adapun gambaran umum dari perancangan pesan telah lulus latihan pada aplikasi *mobile* "Latihan Daya Ingat" dapat dilihat pada Gambar 3.45.

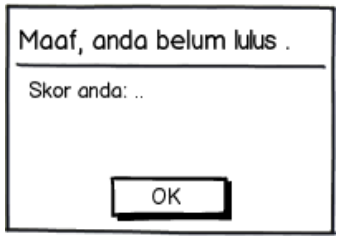


Gambar 3.45 Perancangan Pesan Telah Lulus Latihan

5. Perancangan Pesan Belum Lulus Latihan

Pesan belum lulus latihan berisi informasi bahwa pengguna belum lulus latihan beserta skor yang diperoleh. Pesan tersebut muncul pada saat pengguna telah menyelesaikan latihan, tetapi belum lulus. Adapun gambaran umum dari perancangan pesan belum lulus latihan pada aplikasi *mobile* "Latihan Daya Ingat" dapat dilihat pada Gambar 3.46.

P05



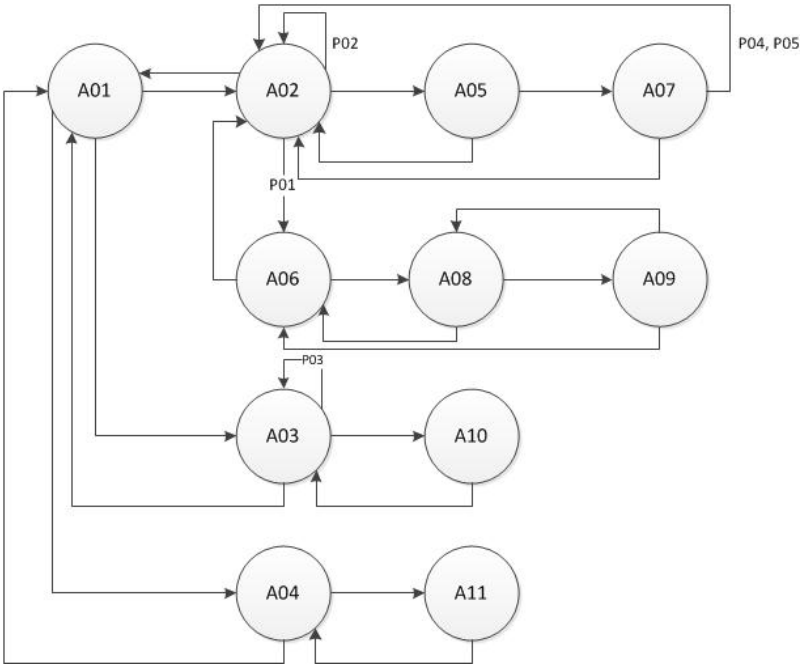
- Klik tombol "OK" untuk menutup pesan dan menuju ke A02.

- Ukuran pesan menyesuaikan.
- Font, ukuran, dan warna tulisan menyesuaikan.

Gambar 3.46 Perancangan Pesan Belum Lulus Latihan

3.2.5 Jaringan Semantik

Jaringan semantik merupakan gambaran pengetahuan grafis yang menunjukkan hubungan antar objek (lingkaran) dan garis yang menggambarkan formasi antar objek tersebut.



Gambar 3.47 Jaringan Semantik Aplikasi "Latihan Daya Ingat"