

BAB 1

SPEKIFIKASI KEBUTUHAN PERANGKAT LUNAK

1.1 Pendahuluan

Universitas YARSI telah menghadirkan Perpustakaan Islam Digital (PID) Al-Maktabah berbasis web pada tahun 2022 (Yuhari, 2022). Web Maktabah YARSI sudah dilengkapi dengan berbagai fitur inovatif, pengguna merasa perlu adanya kemampuan untuk mengunduh aplikasi Maktabah ke perangkat Android mereka. Aplikasi *mobile*, sebagai sistem operasi *mobile* berbasis Linux, menjadi pilihan ideal untuk mendukung pengembangan aplikasi ini (Supardi, 2017). Pengembang menggunakan *Web Service* meningkatkan fungsionalitas aplikasi, khususnya RESTful API, yang terdiri dari berbagai *endpoint* dengan fungsionalitas yang terdefinisi (Ehsan *et al.*, 2022).

Perkembangan penggunaan perangkat seluler Android, aplikasi khusus Maktabah YARSI untuk platform ini menjadi sangat penting. Solusi ini diharapkan dapat memberikan kepraktisan bagi pengguna dalam mengakses sumber daya perpustakaan digital, seperti buku, jurnal, dan makalah, langsung dari ponsel pintar mereka. Pengembangan aplikasi *mobile* ini memberikan sejumlah manfaat praktis, di mana pengembang diberikan fleksibilitas dalam mendistribusikan aplikasi sesuai dengan skema lisensi yang mereka pilih (Safaat, 2011).

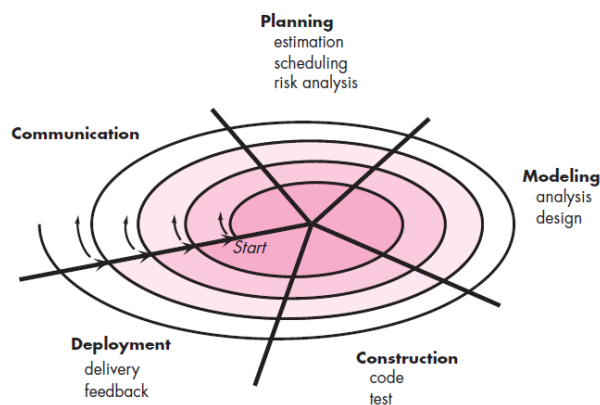
Aplikasi android *mobile* Maktabah YARSI memberikan fitur pencarian dan navigasi yang mudah digunakan, meningkatkan pengalaman pengguna dalam menjelajahi koleksi perpustakaan. Fitur Masuk dan Daftar dalam aplikasi ini memungkinkan setiap pengguna untuk menandai buku favorit mereka, mempersingkat waktu dalam mencari kembali buku yang pernah dibaca. Keseluruhan, pengembangan aplikasi *mobile* ini harus diprioritaskan untuk memenuhi kebutuhan pengguna dalam mengakses Maktabah YARSI secara lebih praktis dan nyaman. Integrasi *Web Service* yang digunakan aplikasi ini tidak hanya memberikan kenyamanan lokal tetapi juga dapat diakses secara daring, menciptakan ekosistem informasi yang lebih terhubung dan terintegrasi.

1.1.1 Siklus Hidup Pengembangan Aplikasi

Upaya menghasilkan aplikasi unggulan, tim pengembang Maktabah YARSI memilih Metodologi Spiral (Gambar 1) sebagai landasan utama. Metodologi Spiral dipilih karena memberikan fleksibilitas dan adaptabilitas yang tinggi, sangat sesuai untuk proyek-proyek kompleks seperti pembangunan aplikasi ini.

Proses pengembangan proyek dimulai tim melakukan komunikasi wawancara narasumber yang terdiri dari Bapak Aya Yahya Maulana, Lc., M.H, Ibu Herika Hayurani, S.Kom., M.Kom, Bapak Muhammad Rizky Yuhari, S.Kom, Bapak Muhammad Raihan Suryanom, S.Kom, Bapak Agus Rifai, Ph.D, dan Bapak Tiko Dhafin Rizky, S.IP., M.M.. Hasil wawancara yang dilakukan dapat mencakup identifikasi kebutuhan dan tujuan proyek serta melibatkan pemangku kepentingan untuk memahami harapan dan ekspektasi pengguna. Selanjutnya, tahap perencanaan dilakukan dengan menyusun rencana pengembangan yang mencakup alokasi sumber daya dan jadwal waktu, serta menganalisis risiko-risiko potensial yang mungkin muncul selama proses pengembangan. Pada tahap pemodelan, arsitektur sistem dan desain aplikasi ditentukan, bersama dengan Identifikasi kebutuhan fungsional dan non-fungsional. Proses dilanjutkan dengan konstruksi, di mana implementasi dan pengkodean dilakukan berdasarkan desain yang telah disetujui, dan uji coba unit dilakukan untuk memastikan bahwa setiap bagian berfungsi dengan baik. Akhirnya, pada tahap *deployment*, aplikasi diimplementasikan dan disebarkan ke lingkungan produksi, diikuti dengan pemantauan performa dan tanggapan terhadap umpan balik pengguna.

Bab ini fokus akan diberikan pada tahapan awal pengembangan, yaitu analisis kebutuhan perangkat lunak atau yang sering disebut sebagai spesifikasi kebutuhan perangkat lunak. Analisis ini melibatkan identifikasi kebutuhan pengguna, fungsi-fungsi utama aplikasi, dan batasan-batasan yang perlu diperhatikan selama proses pengembangan.



Gambar 1. Metodologi Penelitian Spiral

Melalui penerapan Metodologi Spiral, diharapkan Aplikasi android *mobile* Maktabah YARSI dapat dikembangkan secara sistematis, meminimalkan risiko, dan memastikan bahwa setiap tahapan pengembangan dapat dijalankan dengan optimal sesuai dengan kebutuhan proyek.

1.1.2 Tujuan Penulisan Spesifikasi Kebutuhan Perangkat Lunak

Spesifikasi Kebutuhan Perangkat Lunak (SKPL) untuk Maktabah YARSI bertujuan memberikan panduan teknis bagi pengembang dalam setiap tahap pengembangan, memastikan keseragaman, dan menyusun kriteria pengukuran kinerja. Bagi pengguna, SKPL menjamin pemenuhan kebutuhan dan harapan, memberikan antarmuka pengguna yang ramah, kejelasan fungsionalitas, dan layanan berkualitas tinggi. SKPL diharapkan dapat mendukung pengembangan efisien Aplikasi android *mobile* Maktabah YARSI, memberikan pengalaman pengguna optimal dalam mengakses dan mengelola sumber daya perpustakaan Islam digital.

1.1.3 Cakupan Produk

Maktabah YARSI yang dikembangkan akan digunakan untuk membaca buku Islam secara digital. Pada tahun 2022 sudah dikembangkan Maktabah YARSI berbasis web (Suryanom, 2022). Fitur yang dibutuhkan pengguna adalah kemampuan mengunduh Aplikasi android *mobile* Maktabah YARSI ke ponsel pintar Android mereka. Android merupakan sistem operasi *mobile* berbasis linux yang diantaranya memiliki sistem operasi, *middleware*, dan aplikasi (Supardi, 2017). Tidak hanya Android untuk menunjang pengembangan aplikasi, tapi juga menggunakan *Web Service* salah satunya yaitu RESTful API. RESTful API atau yang biasa dikenal sebagai web API terdiri dari

endpoint. Setiap *endpoint*-nya memiliki fungsionalitas dari proses bisnis. API ini biasanya diakses menggunakan HTTP yang sudah terdefinisi seperti GET, POST, PUT, DELETE (Ehsan *et al.*, 2022). Seiring dengan kemajuan teknologi, pengguna saat ini semakin mengandalkan perangkat seluler Android mereka untuk mengakses berbagai layanan secara praktis dan cepat. Aplikasi Android khusus untuk Perpustakaan Maktabah YARSI, pengguna akan mendapatkan sejumlah manfaat penting dengan praktis.

Untuk mengatasi permasalahan yang sudah dijabarkan, Aplikasi android *mobile* Maktabah YARSI hadir menjadi solusi untuk mempermudah pengguna dalam mengakses berbagai sumber daya perpustakaan digital seperti buku, jurnal, makalah, dan koleksi lainnya, langsung dari ponsel pintar mereka. Dengan adanya aplikasi android *mobile* Maktabah YARSI, pengguna akan memiliki alat pencarian dan navigasi yang lebih mudah digunakan, yang akan meningkatkan pengalaman mereka dalam menjelajahi koleksi perpustakaan. Fitur ini akan membantu pengguna menemukan informasi yang mereka butuhkan dengan lebih efisien. Aplikasi ini memiliki fitur favorit, sehingga setiap pengguna bisa menandai buku yang disukai dan yang ingin dibaca lagi. Aplikasi ini memiliki fitur menandai buku sebagai favorit, sehingga membuat pengguna lebih nyaman dan lebih cepat untuk mencari buku yang sebelumnya pernah dibaca dan ditandai. Namun, fitur tersebut bisa diakses khusus pengguna yang sudah melakukan daftar dan masuk ke akun mereka.

Dengan mengembangkan Aplikasi android *mobile* Maktabah YARSI berbasis *mobile*, sejalan dengan firman Allah ﷻ dalam surah Al-Alaq (96): 1-5

اِقْرَأْ بِاسْمِ رَبِّكَ الَّذِي خَلَقَ الْإِنْسَانَ مِنْ عَلَقٍ اِقْرَأْ وَرَبُّكَ الْأَكْرَمُ الَّذِي عَلَّمَ بِالْقَلَمِ
عَلَّمَ الْإِنْسَانَ مَا لَمْ يَعْلَمْ

Artinya: “Bacalah dengan (menyebut) nama Tuhanmu Yang Menciptakan. Dia telah menciptakan manusia dari segumpal darah. Bacalah, dan Tuhanmulah Yang Maha Pemurah, Yang mengajar (manusia) dengan perantaraan qalam. Dia mengajarkan kepada manusia apa yang tidak diketahuinya”. (QS. Al-Alaq (96): 1-5)

Al-Qur'an dalam surah Al-Alaq (96): 1-5 menekankan pentingnya membaca dengan menyebut nama Tuhan Yang Menciptakan. Firman Allah ﷻ tersebut menggambarkan proses penciptaan manusia dari segumpal darah, memberikan penekanan pada perintah untuk membaca sebagai awal dari pengajaran Allah ﷻ kepada manusia.

Tafsir Ibn Katsir memberikan pemahaman mendalam tentang ayat-ayat tersebut. Al-Qur'an, sebagai rahmat pertama yang Allah ﷻ turunkan, menunjukkan kasih sayang-Nya kepada manusia. Ayat-ayat tersebut dianggap sebagai awal dari petunjuk Allah ﷻ kepada hamba-Nya. Kasih sayang Allah ﷻ termanifestasi dalam pengajaran-Nya kepada manusia, yang dimulai dengan pengetahuan bahwa manusia berasal dari segumpal darah (Muhammad, 2022).

Ibn Katsir juga menekankan pentingnya ilmu diin (ilmu agama) dalam meningkatkan kehormatan seseorang. Ilmu agama membedakan manusia, seperti Adam, dari para malaikat. Ilmu dapat bersifat pemikiran, lisan, maupun tertulis, dan dianggap sebagai sarana untuk menyampaikan dan menyebarkan pemikiran (Muhammad, 2022).

Konsep mengikat ilmu dengan tulisan, seperti yang disebutkan dalam atsar, menunjukkan pentingnya dokumentasi dan penyebaran ilmu. Keterlibatan tulisan tangan dalam menyampaikan pemikiran dan ilmu menjadi bagian dari upaya untuk menyebarkan pengetahuan.

Peradaban Islam juga menekankan pentingnya ilmu. Rasulullah ﷺ memberikan perhatian khusus kepada mereka yang memiliki ilmu, dan ajaran-ajaran Islam menggarisbawahi nilai tinggi ilmu pengetahuan dalam memajukan peradaban. Teknologi modern melalui Aplikasi android *mobile* Maktabah YARSI diharapkan dapat memperluas akses terhadap ilmu pengetahuan dan memenuhi tuntutan zaman dengan sejalan dengan nilai-nilai Islam.

Selain surah Al-Alaq (96): 1-5, terdapat juga potongan surah Surah Al-Maidah (5): 2

... وَتَعَاوَنُوا عَلَى الْبِرِّ وَالتَّقْوَىٰ ...

Artinya: "... Dan tolong menolonglah kamu dalam (mengerjakan) kebajikan dan takwa..." (QS. Al-Maidah (5): 2)

Aplikasi android *mobile* Maktabah YARSI dianggap sebagai sarana yang mendukung kerjasama dalam mencapai tujuan tersebut. Aplikasi ini disajikan sebagai alat untuk memfasilitasi umat Islam dalam meningkatkan pemahaman dan pengetahuan mereka terhadap ajaran agama Islam.

Tafsir Al-Muyassar menguraikan pesan dari ayat 2 surah Al-Maidah tersebut. Dalam konteks historis awal Islam, tafsir ini mengingatkan umat Islam untuk mematuhi batasan dan rambu-rambu yang ditetapkan Allah ﷻ. Ditegaskan agar tidak berperang di bulan-bulan haram, seperti Dzul Qa'dah, Dzul Hijjah, Muharram, dan Rajab. Disarankan untuk tidak menghalalkan hewan hadyu dan hewan yang sudah dikalungi kalung, yang merupakan tanda pelaksanaan ibadah haji (Umar, 2016).

Tolong-menolong di antara umat Islam sangat penting untuk melaksanakan kebaikan dan takwa juga dijelaskan. Tafsir ini menegaskan bahwa umat Islam seharusnya bekerja sama dalam memperbaiki kehidupan mereka dan mencari karunia Allah ﷻ. Namun, peringatan diberikan agar kolaborasi ini tidak terjadi dalam dosa, kemaksiatan, atau pelanggaran terhadap hukum-hukum Allah ﷻ.

Pesan tersebut mencerminkan prinsip-prinsip keadilan, saling menghormati, dan mematuhi aturan Allah ﷻ. Selain itu, tafsir ini mengingatkan umat Islam untuk tidak membiarkan kebencian terhadap suatu kelompok menghalangi mereka untuk berlaku adil. Dalam segala bentuk kerjasama, umat Islam diingatkan untuk selalu mengutamakan kebaikan dan takwa, serta menjauhi dosa dan kemaksiatan, sebagai bentuk taat kepada perintah Allah ﷻ yang memiliki konsekuensi siksa yang keras jika diabaikan.

Dalam surah Al-Insyirah (94): 5-6, Allah ﷻ menyampaikan pesan yang mengandung hikmah besar.

فَإِنَّ مَعَ الْعُسْرِ يُسْرًا , إِنَّ مَعَ الْعُسْرِ يُسْرًا

Artinya: “Maka sesungguhnya bersama kesulitan ada kemudahan. Sesungguhnya bersama kesulitan ada kemudahan.” (QS. Al-Insyirah (94): 5-6)

Ayat ini menjadi inspirasi bagi pengembangan Aplikasi android *mobile* Maktabah YARSI berbasis *mobile*. Aplikasi tersebut diharapkan dapat memberikan kemudahan kepada pengguna dalam mengakses berbagai sumber daya perpustakaan digital. Teknologi yang dimanfaatkan untuk aplikasi ini menjadi sarana yang efisien untuk menyebarkan ilmu pengetahuan kepada umat Islam di seluruh dunia.

Penjelasan Quraish Shihab dalam tafsir al-Misbah menyoroti interpretasi ulama terhadap kata "مع" (ma'a) dalam ayat tersebut. Harfiahnya bermakna "bersama," beberapa ulama

memahami bahwa dalam konteks ayat ini, maknanya dapat diartikan sebagai "sesudah." Tafsir ini memperkuat pemahaman bahwa kemudahan akan datang setelah melewati kesulitan (Harfin, 2021).

Az-Zamakhshari juga menegaskan konsep dekatnya waktu antara kemudahan dan kesulitan. Penggunaan kata "bersama" disini tidak hanya menunjukkan keterkaitan antara keduanya, tetapi juga mencerminkan betapa cepatnya perubahan dari keadaan sulit menjadi mudah yang akan dialami oleh individu atau umat.

Aplikasi android *mobile* Maktabah YARSI diharapkan menjadi salah satu bentuk kemudahan yang Allah ﷻ janjikan, sebagaimana tercantum dalam surah Al-Insyirah. Penggunaan teknologi dan penyebaran ilmu pengetahuan, aplikasi ini dapat menjadi wadah untuk memberikan manfaat kepada umat Islam, sejalan dengan ajaran suci Al-Qur'an.

Hadis dari At-Tirmidzi memperkuat dan mengukuhkan pentingnya ilmu dan proses pembelajaran dalam Islam. Hadis tersebut menggambarkan ajaran Rasulullah ﷺ yang memberikan nilai tinggi pada usaha mencari ilmu yang secara harfiah berarti bahwa seseorang yang berusaha mencari ilmu dianggap sedang berada dalam perjalanan yang dijalannya demi Allah ﷻ. Rasulullah ﷺ bersabda:

مَنْ خَرَجَ فِي طَلَبِ الْعِلْمِ فَهُوَ فِي سَبِيلِ اللَّهِ حَتَّى يَرْجِعَ

Artinya: "*Barang siapa keluar mencari ilmu, maka dia berada di jalan Allah ﷻ hingga kembali.*" (Riwayat At-Tirmidzi).

Pernyataan ini menekankan bahwa pencarian ilmu dianggap sebagai bentuk ibadah dan jihad yang dijalankan untuk mendekatkan diri kepada Allah ﷻ. Kata lain yang biasa digunakan yaitu setiap langkah yang diambil dalam upaya mencari ilmu dianggap sebagai langkah yang dijalani di jalan Allah ﷻ.

Nilai-nilai yang dipertimbangkan, terkandung dalam ayat-ayat Al-Qur'an dan hadis di atas, Aplikasi android *mobile* Maktabah YARSI diarahkan untuk menjadi sarana yang memfasilitasi umat Islam dalam menyumbangkan ilmu pengetahuan. Aplikasi ini diharapkan dapat memberikan akses mudah dan luas kepada berbagai sumber ilmu pengetahuan, sejalan dengan prinsip-prinsip ajaran Islam yang mendorong pencarian ilmu

sebagai suatu bentuk ibadah. Aplikasi android *mobile* Maktabah YARSI dapat dianggap sebagai wadah yang mendukung umat Islam dalam berinfak ilmu pengetahuan, menyediakan platform untuk peningkatan pengetahuan dan pemahaman terhadap ajaran agama Islam.

Selain itu, Surah Al-Mujadila (58): 11 menegaskan pentingnya pengetahuan dalam masyarakat Islam:

يَا أَيُّهَا الَّذِينَ آمَنُوا إِذَا قِيلَ لَكُمْ تَفَسَّحُوا فِي الْمَجَالِسِ فَافْسَحُوا يَفْسَحِ اللَّهُ لَكُمْ وَإِذَا قِيلَ
انشُزُوا فَانْشُزُوا يَرْفَعِ اللَّهُ الَّذِينَ آمَنُوا مِنْكُمْ وَالَّذِينَ أُوتُوا الْعِلْمَ دَرَجَاتٍ وَاللَّهُ بِمَا
تَعْمَلُونَ خَبِيرٌ

Artinya: "Wahai orang-orang yang beriman, apabila dikatakan kepadamu, 'Berilah jalan di majlis-majlis', maka berilah jalanlah, niscaya Allah ﷻ akan memberi kelapangan untukmu. Dan apabila dikatakan, 'Berdirilah', maka berdirilah, niscaya Allah ﷻ akan meninggikan orang-orang yang beriman di antaramu dan orang-orang yang diberi ilmu pengetahuan beberapa derajat. Dan Allah ﷻ Maha Mengetahui apa yang kamu kerjakan." (QS. Al-Mujadila (58): 11).

Dalam penjelasan Tafsir Kementerian Agama, Surah Al-Mujadila ayat 11 diartikan sebagai petunjuk bahwa jika Rasulullah ﷺ memerintahkan sebagian kaum Muslimin untuk memberikan kesempatan kepada orang tertentu untuk duduk, atau mereka diinstruksikan untuk pergi terlebih dahulu, maka seharusnya mereka patuh dengan berdiri atau pergi. Hal ini dilakukan sebagai bentuk penghormatan terhadap keinginan Rasulullah ﷺ yang ingin menyendiri untuk merenungkan urusan-urusan agama atau melaksanakan tugas-tugas yang memerlukan perhatian khusus (Permana & Yulianto, 2023).

Dengan merujuk pada ayat ini, Aplikasi android *mobile* Maktabah YARSI dapat dianggap sebagai salah satu cara untuk memberikan "jalan" atau akses bagi umat Islam dalam memperluas pengetahuan dan meningkatkan kedudukan di mata Allah ﷻ melalui ilmu pengetahuan.

Dengan demikian, pengembangan Aplikasi android *mobile* Maktabah YARSI tidak hanya memenuhi kebutuhan praktis pengguna tetapi juga sejalan dengan nilai-nilai Islam yang mendorong umatnya untuk mencari ilmu pengetahuan dan menyebarkannya untuk kebaikan bersama.

1.1.4 Definisi, Singkatan, dan Akronim

Berikut adalah definisi, singkatan, dan akronim dari beberapa istilah yang telah disebutkan:

Tabel 1. Akronim, Singkatan, Deskripsi Bab 1

Akronim	Singkatan	Deskripsi
Android	-	Sistem operasi yang populer di kalangan pelajar dan banyak digunakan oleh pengembang perangkat lunak untuk menciptakan inovasi-inovasi baru. (Suwarti & Catriwati, 2022).
API	<i>Application Programming Interface</i>	Dokumentasi yang mencakup rincian antarmuka, fungsi, kelas, struktur, dan elemen-elemen lain yang diperlukan untuk mengembangkan perangkat lunak. (Novianto, 2015).
Dashboard	-	Alat visualisasi yang menampilkan grafis dari informasi kunci yang diperlukan untuk mencapai satu atau lebih tujuan. (Sihombing <i>et al.</i> , 2019).
HTTP	Hypertext Transfer Protocol	Sebuah protokol meminta atau menjawab antara

Akronim	Singkatan	Deskripsi
		<i>client</i> dan server (Zabar & Novianto, 2015).
<i>Mobile</i>	-	Aplikasi perangkat lunak yang dapat dijalankan di perangkat <i>mobile</i> . (<i>Smartphone</i> , <i>Tablet</i> , <i>iPod</i> , dan lain-lain).
Nest JS	-	<i>Framework</i> yang dibuat untuk mempermudah developer di bagian <i>back-end</i> (Arif, 2019).
PID	Perpustakaan Islam Digital	Kumpulan bahan pustaka keislaman (teks, gambar, video, dan lain-lain.) yang disimpan dalam format digital dan dapat diakses melalui beberapa media, termasuk jaringan komputer, terutama Internet, baik memiliki perpustakaan secara fisik ataupun tidak (Maulana <i>et al.</i> , n.d.).
RESTful API	<i>Representational State Transfer Application Programming Interface</i>	Tipe arsitektur dari <i>Application Programming Interface</i> atau API (Kaniya <i>et al.</i> , 2022).
SKPL	Spesifikasi Kebutuhan Perangkat Lunak	Detail tentang kemampuan dan fungsi yang diinginkan serta batasan, karakteristik,

Akronim	Singkatan	Deskripsi
		kualitas, dan properti yang dibutuhkan oleh perangkat lunak yang akan dikembangkan, berdasarkan persyaratan dan kebutuhan yang harus dipenuhi oleh perangkat lunak atau diinginkan oleh pengguna (Manek, 2019).
<i>Web Service</i>	-	Aplikasi yang di dalamnya berisi kumpulan basis data atau database (Olif, 2023).

1.1.5 Deskripsi Umum Bab

SKPL untuk Aplikasi android *mobile* Maktabah YARSI ini menjelaskan sistematika dari perangkat lunak sebagai berikut:

- **Pendahuluan**

Pada bagian ini, akan dijelaskan permasalahan yang ingin diatasi oleh Aplikasi android *mobile* Maktabah YARSI. Ini mencakup gambaran umum mengenai kebutuhan pengguna, konteks penggunaan, dan lingkungan aplikasi. Informasi tambahan dapat melibatkan tujuan pembangunan aplikasi, serta standar atau regulasi yang perlu diperhatikan.

- **Deskripsi Umum Perangkat Lunak**

Pada bagian ini, akan dijelaskan konteks dan asal-usul dari Aplikasi android *mobile* Maktabah YARSI. Ini termasuk penjelasan mengenai pemilihan teknologi, platform, dan arsitektur sistem yang digunakan.

- **Kebutuhan Fungsional**

Pada bagian ini, akan diuraikan kebutuhan fungsional Aplikasi android *mobile* Maktabah YARSI. Ini mencakup fitur-fitur yang diharapkan oleh pengguna dan diperoleh melalui wawancara serta analisis kebutuhan.

- **Fitur Sistem**

Pada bagian ini, akan dijelaskan cara mengorganisir kebutuhan fungsional ke dalam fitur-fitur sistem. Ini mencakup layanan utama yang disediakan oleh Aplikasi android *mobile* Maktabah YARSI untuk memenuhi kebutuhan pengguna.

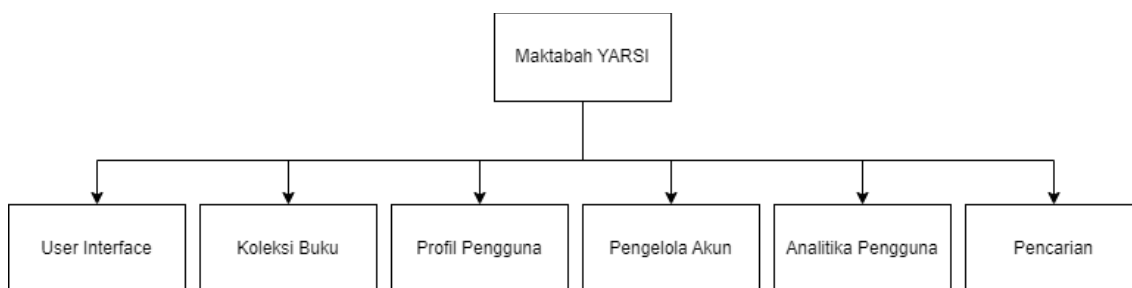
- **Kebutuhan Non Fungsional**

Pada bagian ini, akan dijelaskan atribut kualitas dalam Aplikasi android *mobile* Maktabah YARSI beserta batas minimum yang dapat diterima dari setiap atribut kualitas. Ini melibatkan aspek-aspek seperti keamanan, kinerja, dan usabilitas.

1.2 Deskripsi Umum Perangkat Lunak

1.2.1 Perspektif Produk

Aplikasi android *mobile* Maktabah YARSI adalah perangkat lunak yang digunakan Dosen, Mahasiswa, maupun masyarakat di luar Universitas YARSI. Aplikasi android *mobile* Maktabah YARSI dapat memudahkan pengguna untuk membaca buku-buku Islam yang telah diwaqafkan. Pengembangan aplikasi ini melanjutkan pengembangan yang sebelumnya sudah ada versi web, dan *dashboard*. Diagram utama dari sistem dapat dilihat pada Gambar 2.



Gambar 2. Diagram Maktabah YARSI

1.2.2 Fungsi Produk

Aplikasi android *mobile* Maktabah YARSI dibuat untuk membantu Perpustakaan Universitas YARSI yang memfasilitasi mahasiswa untuk membaca buku. Fungsi utama dari aplikasi ini sebagai berikut:

1. Pengguna dapat membaca buku-buku Islam yang sudah dibagi sesuai kategori;
2. Pengguna dapat menyimpan buku favorit dalam aplikasi (bila sudah masuk);
3. Pengguna dapat melihat riwayat buku yang pernah dibaca (bila sudah masuk);

4. Pengguna dapat mencari buku yang tersedia di fitur pencarian;
5. Pengguna dapat mengganti profil (bila sudah masuk).

1.2.3 Kelas dan Karakteristik Pengguna

Pengguna Aplikasi android *mobile* Maktabah YARSI terbagi menjadi dua karakteristik yaitu pengguna dan pengelola data.

1. Pengguna
Hak Akses: Membaca, mencari, menambah buku ke dalam favorit, dan mengubah profil.
2. Pengelola Data
Hak Akses: Menambah, menghapus, menyunting aset Aplikasi android *mobile* Maktabah YARSI.

1.2.4 Lingkungan Operasi

Aplikasi android *mobile* Maktabah YARSI akan digunakan oleh dosen, mahasiswa, dan masyarakat umum. Aplikasi android *mobile* Maktabah YARSI dapat berjalan pada platform Android dengan spesifikasi minimal menggunakan Android versi 7.

1.2.5 Batasan Perancangan dan Implementasi

Aplikasi android *mobile* Maktabah YARSI dapat digunakan setiap saat. Aplikasi ini terdapat beberapa batasan antara lain:

1. Aplikasi android *mobile* Maktabah YARSI hanya dapat digunakan pada platform Android
2. Aplikasi android *mobile* Maktabah YARSI dikembangkan menggunakan Visual Studio Code dengan bahasa pemrograman Nest JS dan Android Studio dengan bahasa pemrograman Kotlin.

1.2.6 Dokumentasi Pengguna

Dokumentasi pengguna Aplikasi android *mobile* Maktabah YARSI dapat dijabarkan secara ringkas ke dalam dua belas (12) fitur sebagai berikut:

1. Tamu

- Seorang Tamu dapat mengakses aplikasi android *mobile* Maktabah YARSI dengan mengklik tombol “masuk tamu” pada halaman masuk. Pengguna Tamu tidak perlu daftar dan masuk dengan akun tertentu.

2. Daftar

- Pengguna dapat mengakses fitur lebih lengkap dari pada Tamu dengan melakukan pendaftaran akun terlebih dahulu. Untuk membuat akun Aplikasi android *mobile* Maktabah YARSI, pengguna mengklik tombol “daftar” pada halaman masuk. Pada form pendaftaran akun, pengguna harus memasukkan *username*, *e-mail*, *password*, dan konfirmasi *password*.

3. Masuk

- Setelah pengguna membuat akun, pengguna dapat masuk ke aplikasi menggunakan data akun berupa *e-mail* dan *password* yang sebelumnya sudah didaftarkan.

4. Pengaksesan *Home*

- Setelah berhasil masuk, pengguna dapat melihat halaman *Home* Aplikasi android *mobile* Maktabah YARSI. Di *Home* terlihat fitur untuk pencarian, jumlah pengunjung aplikasi, kategori buku, jumlah buku, buku terbaru, rekomendasi buku, iklan buku.

5. Kategori dan Sub Kategori

- Pada halaman *Home*, pengguna dapat mengakses kategori dan sub kategori buku dengan mengklik kategori terlebih dahulu, lalu mengklik sub kategori. Setelah mengklik sub kategori, pengguna dapat melihat *list*/daftar buku yang terdapat pada sub kategori yang telah dipilih.

6. List buku

- Setelah masuk ke *list* buku, pengguna memilih buku kemudian akan masuk ke halaman detail buku.

7. Detail Buku

- Setelah masuk ke detail buku, pengguna dapat melihat poster buku, judul buku, jumlah terbaca, jumlah halaman, tahun terbit, dan deskripsi buku. Pada halaman detail buku, pengguna dapat membaca buku dan/atau menandai buku sebagai favorit.

8. Isi Buku

- Setelah masuk ke isi buku, pengguna dapat membaca buku dan melihat daftar isi.

9. Riwayat

- Pengguna dapat melihat buku yang terakhir dibaca pada menu Riwayat.

10. Favorit

- Pengguna dapat melihat buku yang sudah ditandai sebagai favorit pada menu Favorit.

11. Saya

- Pada menu Saya, pengguna dapat melakukan beberapa hal seperti mengganti tema, melihat tentang aplikasi, atau mengedit profil pengguna.

12. Keluar

- Setelah selesai menggunakan aplikasi, pengguna dapat keluar dengan menekan tombol keluar pada bagian bawah.

Tim pengembang telah membuat buku manual yang bisa digunakan sebagai panduan dalam menggunakan aplikasi *mobile* android Maktabah YARSI. Dokumentasi buku manual dapat diakses pada tautan : <https://bit.ly/ManualBook-MaktabahYARSI>

1.2.7 Asumsi dan Dependensi

Faktor asumsi dan dependensi pada Aplikasi android *mobile* Maktabah YARSI adalah sebagai berikut:

1. Asumsi

- Aplikasi android *mobile* Maktabah YARSI dapat digunakan kapanpun dan dimanapun
- Pengembangan akan memakan waktu lebih lama dan desain akan menyesuaikan tampilan di *mobile*
- Aplikasi android *mobile* Maktabah YARSI dapat dijalankan pada *mobile* dengan spesifikasi minimal Android 7

2. Dependensi

- Aplikasi ini menggunakan bahasa Kotlin untuk *mobile*, dan Nest JS untuk *back-end*;
- Aplikasi ini menggunakan MongoDB sebagai *Database* Aplikasi android *mobile* Maktabah YARSI;

1.3 Kebutuhan Fungsional

1.3.1 Antarmuka Pengguna

Pengguna berinteraksi dengan Aplikasi android *mobile* Maktabah YARSI melalui antarmuka grafis atau *Graphical User Interface* (GUI). Aplikasi android *mobile* Maktabah YARSI menampilkan menu utama dan berbagai sumber daya perpustakaan digital seperti buku, jurnal, dan makalah melalui layar ponsel pintar pengguna. Pengguna dapat memberikan masukan melalui sentuhan layar pada ponsel pintar Android mereka untuk menjelajahi koleksi perpustakaan.

1.3.2 Antarmuka Perangkat Keras

Kebutuhan minimum perangkat keras yang digunakan pemakai untuk berinteraksi dengan Aplikasi android *mobile* Maktabah YARSI adalah:

1. RAM: Minimum 1 GB untuk kinerja yang optimal.
2. Penyimpanan: Setidaknya 100 MB ruang penyimpanan untuk instalasi dan caching data.
3. Layar: Resolusi layar yang mendukung tampilan grafis yang baik.
4. Koneksi Internet: Diperlukan untuk mengakses buku dan sumber daya *online*.

1.3.3 Antarmuka Perangkat Lunak

Aplikasi android *mobile* Maktabah YARSI merupakan solusi terintegrasi yang tidak berdiri sendiri, melainkan terkoneksi dengan sejumlah komponen perangkat lunak untuk menyediakan layanan perpustakaan digital yang efisien dan komprehensif. Aplikasi *mobile* android Maktabah YARSI Dirancang khusus untuk berjalan pada perangkat Android dengan sistem operasi versi 7.0 (*Nougat*) ke atas. Aplikasi ini memanfaatkan sejumlah elemen perangkat lunak, termasuk sistem manajemen *database* yang mendukung struktur data buku dan pengguna. Adapun keterhubungan dengan *library* dan *framework* Android, seperti Android Support Library, yang memberikan keleluasaan dalam pengembangan antarmuka pengguna yang responsif. Aplikasi ini juga melakukan komunikasi dengan server melalui protokol API, dengan implementasi RESTful API yang memfasilitasi pertukaran data yang efisien. Dalam hal data dan pesan, aplikasi menerima permintaan pengguna seperti pencarian buku atau pembelian, dan memberikan respons yang mencakup informasi buku dan status transaksi. Layanan yang dibutuhkan, seperti pencarian buku dan pengelolaan akun, dilakukan melalui komunikasi *HyperText*

Transfer Protocol Secure (HTTPS) untuk memastikan keamanan. Meskipun memiliki ketergantungan pada koneksi internet, Aplikasi android *mobile* Maktabah YARSI memberikan pengalaman pengguna yang memuaskan dan memudahkan akses terhadap berbagai sumber daya perpustakaan digital dengan efisien.

1.3.4 Antarmuka Komunikasi

Aplikasi android *mobile* Maktabah YARSI menggunakan standar komunikasi HTTP/HTTPS saat mengambil data dari RESTful API perpustakaan digital. Penggunaan HTTPS memastikan keamanan data selama pertukaran informasi. Mekanisme sinkronisasi diaktifkan saat mengambil data dari API untuk memperbarui informasi seperti koleksi buku dan jurnal. Pendekatan ini memungkinkan aplikasi berfungsi efisien, mengoptimalkan komunikasi dengan server hanya saat dibutuhkan, memastikan akses pengguna ke informasi terbaru dari perpustakaan digital.

1.4 Fitur Sistem

1.4.1 Fitur *Splash Screen*

1.4.1.1 Deskripsi

Fitur *Splash Screen* pada Aplikasi android *mobile* Maktabah YARSI adalah tampilan awal yang muncul sejenak ketika pengguna membuka aplikasi untuk pertama kalinya. Tampilan ini berisi logo Maktabah YARSI untuk memberikan kesan visual sebelum masuk ke halaman utama.

1.4.1.2 *Trigger*

Fitur ini diaktifkan secara otomatis ketika pengguna membuka Aplikasi android *mobile* Maktabah YARSI.

1.4.1.3 *Input*

Fitur *Splash Screen* tidak melibatkan *input* dari pengguna. Data yang ditampilkan bersifat statis dan terintegrasi langsung dalam kode aplikasi.

1.4.1.4 *Output*

Output dari Fitur *Splash Screen* adalah tampilan visual yang muncul di layar pengguna selama beberapa detik saat aplikasi pertama kali dibuka.

1.4.1.5 Skenario Utama

Prakondisi: Aplikasi android *mobile* Maktabah YARSI telah diunduh dan diinstal pada perangkat pengguna.

Pascakondisi: Pengguna melihat tampilan *Splash Screen* sebelum masuk ke tampilan utama aplikasi.

Langkah-langkah:

1. Pengguna membuka Aplikasi android *mobile* Maktabah YARSI.
2. *Splash Screen* ditampilkan dengan menampilkan logo dan informasi singkat.
3. Setelah beberapa detik, *Splash Screen* otomatis menghilang, dan pengguna masuk ke halaman utama aplikasi.

1.4.2 Fitur Daftar pada Aplikasi *Mobile*

1.4.2.1 Deskripsi

Fitur Daftar pada Aplikasi android *mobile* Maktabah YARSI memungkinkan pengguna untuk membuat akun baru di dalam aplikasi, memberikan akses ke fitur-fitur khusus yang memerlukan identifikasi pengguna.

1.4.2.2 *Trigger*

Fitur Daftar dapat dieksekusi secara manual oleh pengguna saat pertama kali membuka aplikasi dan memilih opsi "Daftar". *Trigger* otomatis terjadi ketika pengguna menyelesaikan proses pendaftaran.

1.4.2.3 *Input*

Data yang dibutuhkan dalam fitur ini meliputi:

1. *Username*;
2. Alamat *e-mail*;
3. Kata *password*;
4. Konfirmasi *password*

1.4.2.4 *Output*

Luaran dari fitur ini adalah akun pengguna yang terdaftar di dalam sistem aplikasi. Pengguna kemudian dapat melakukan masuk menggunakan informasi yang didaftarkan.

1.4.2.5 Skenario Utama

Prakondisi: Pengguna membuka aplikasi untuk pertama kali.

Pascakondisi: Pengguna berhasil mendaftarkan akun dan diarahkan ke layar Masuk.

Langkah-langkah:

1. Pengguna memilih opsi "Daftar" pada layar awal aplikasi.
2. Sistem menampilkan formulir pendaftaran.
3. Pengguna mengisi formulir dengan *Username*, alamat *e-mail*, *password*, dan konfirmasi *password*.
4. Pengguna menekan tombol "Daftar".
5. Sistem memvalidasi data dan memberikan konfirmasi pendaftaran berhasil.
6. Pengguna diarahkan ke layar masuk.

1.4.2.6 Skenario Eksepsional 1

Prakondisi: Pengguna memasukkan alamat *e-mail* yang sudah terdaftar sebelumnya.

Pascakondisi: Sistem menampilkan pesan kesalahan dan meminta pengguna menggunakan alamat *e-mail* yang berbeda.

Langkah-langkah:

1. Pengguna mengisi formulir pendaftaran dengan alamat *e-mail* yang sudah terdaftar.
2. Pengguna menekan tombol "Daftar".
3. Sistem mendeteksi adanya konflik data dan menampilkan pesan kesalahan.
4. Pengguna diminta untuk menggunakan alamat *e-mail* yang berbeda.

1.4.2.7 Skenario Eksepsional 2

Prakondisi: Pengguna memasukkan *password* dan *Username* yang tidak memenuhi kriteria keamanan.

Pascakondisi: Sistem menampilkan pesan kesalahan dan memberikan panduan untuk memperbaiki *password* dan *Username*.

Langkah-langkah:

1. Pengguna mengisi formulir pendaftaran dengan *password* yang tidak memenuhi kriteria.
2. Pengguna menekan tombol "Daftar".

3. Sistem mendeteksi *password* dan *Username* tidak memenuhi kriteria dan menampilkan pesan kesalahan.
4. Pengguna diminta untuk memperbaiki *password* dan *Username* sesuai panduan yang diberikan.

1.4.3 Fitur Daftar pada Web Service

1.4.3.1 Deskripsi

Fitur Daftar pada *Web Service* Aplikasi android *mobile* Maktabah YARSI memungkinkan untuk membuat akun pengguna baru di server menggunakan layanan web.

1.4.3.2 Trigger

Fitur Daftar pada *Web Service* dapat dipicu oleh Aplikasi android *mobile* Maktabah YARSI ketika pengguna memilih opsi "Daftar" di antarmuka pengguna aplikasi.

1.4.3.3 Input

Data yang diperlukan untuk fitur ini melibatkan informasi pengguna, seperti:

1. *Username*
2. *Alamat e-mail*
3. *Password*
4. *Konfirmasi Password*

1.4.3.4 Output

Luaran dari fitur ini adalah respons dari layanan web yang mencakup status registrasi dan token otentikasi jika registrasi berhasil.

1.4.3.5 Skenario Utama

Prakondisi: Aplikasi *mobile* terbuka dan pengguna memilih opsi "Daftar" di antarmuka pengguna.

Pascakondisi: Pengguna mendapatkan konfirmasi bahwa registrasi berhasil dan akun berhasil dibuat untuk otentikasi selanjutnya.

Langkah-langkah:

1. Pengguna memasukkan informasi registrasi (*Username*, *alamat e-mail*, *konfirmasi password*, dan *password*).
2. Aplikasi *mobile* mengirimkan permintaan registrasi ke layanan web.
3. Layanan web memvalidasi data dan mengembalikan respons registrasi.

4. Jika registrasi berhasil, layanan web menyimpan data pengguna untuk otentikasi selanjutnya.
5. Aplikasi *mobile* menampilkan konfirmasi registrasi berhasil.

1.4.3.6 Skenario Eksepsional 1

Prakondisi: Pengguna mencoba mendaftar dengan alamat *e-mail* yang sudah terdaftar sebelumnya.

Pascakondisi: Layanan web memberikan pesan kesalahan dan memberitahu pengguna untuk menggunakan alamat *e-mail* yang berbeda.

Langkah-langkah:

1. Pengguna mengirimkan permintaan registrasi dengan alamat *e-mail* yang sudah terdaftar.
2. Layanan web mendeteksi konflik data dan mengembalikan pesan kesalahan.
3. Pengguna diberitahu bahwa alamat *e-mail* sudah terdaftar dan diminta menggunakan alamat *e-mail* yang berbeda.

1.4.3.7 Skenario Eksepsional 2

Prakondisi: Pengguna mencoba mendaftar dengan *password* dan *Username* yang tidak memenuhi kriteria keamanan.

Pascakondisi: Layanan web memberikan pesan kesalahan dan memberitahu pengguna untuk memperbaiki *password* dan *Username*.

Langkah-langkah:

1. Pengguna mengirimkan permintaan registrasi dengan *password* dan *Username* yang tidak memenuhi kriteria.
2. Layanan web mendeteksi *password* dan *Username* tidak memenuhi kriteria dan mengembalikan pesan kesalahan.
3. Pengguna diberitahu untuk memperbaiki *password* dan *Username* sesuai panduan yang diberikan.

1.4.4 Fitur Masuk pada Aplikasi Mobile

1.4.4.1 Deskripsi

Fitur Masuk pada Aplikasi android *mobile* Maktabah YARSI untuk pengguna yang telah memiliki akun untuk masuk ke dalam aplikasi, memberikan akses ke fungsi dan konten yang terkait dengan akun pengguna.

1.4.4.2 *Trigger*

Fitur Masuk akan dieksekusi ketika pengguna membuka aplikasi pertama kali setelah di tampilan tutorial pada antarmuka pengguna Aplikasi android *mobile* Maktabah YARSI.

1.4.4.3 *Input*

Data yang diperlukan untuk fitur ini adalah informasi akun pengguna, termasuk:

1. *Username* atau alamat *e-mail*
2. *Password*

1.4.4.4 *Output*

Luaran dari fitur ini adalah akses ke fungsi dan konten yang hanya tersedia untuk pengguna yang telah berhasil masuk ke aplikasi.

1.4.4.5 *Skenario Utama*

Prakondisi: Aplikasi *mobile* terbuka dan pengguna memilih opsi "Masuk" di antarmuka pengguna.

Pascakondisi: Pengguna mendapatkan akses penuh ke fungsi dan konten yang memerlukan otentikasi.

Langkah-langkah:

1. Pengguna memasukkan informasi Masuk (alamat *e-mail*, dan *password*).
2. Aplikasi *mobile* mengirimkan permintaan otentikasi ke server.
3. Server memvalidasi informasi Masuk dan mengembalikan token otentikasi jika berhasil.
4. Aplikasi *mobile* menyimpan token otentikasi untuk otentikasi selanjutnya.
5. Pengguna diberikan akses penuh ke fitur-fitur aplikasi yang memerlukan otentikasi.

1.4.4.6 *Skenario Eksepsional 1*

Prakondisi: Pengguna mencoba Masuk dengan *Username* atau alamat *e-mail* yang tidak terdaftar.

Pascakondisi: Aplikasi memberikan pesan kesalahan dan meminta pengguna memeriksa kembali informasi Masuk.

Langkah-langkah:

1. Pengguna memasukkan alamat *e-mail* yang tidak terdaftar.

2. Server mendeteksi bahwa informasi Masuk tidak valid dan mengembalikan pesan kesalahan.
3. Aplikasi *mobile* menampilkan pesan kesalahan dan meminta pengguna memeriksa kembali informasi Masuk.

1.4.4.7 Skenario Eksepsional 2

Prakondisi: Pengguna mencoba Masuk dengan *password* yang salah.

Pascakondisi: Aplikasi memberikan pesan kesalahan dan meminta pengguna memeriksa kembali *password*.

Langkah-langkah:

1. Pengguna memasukkan *password* yang salah.
2. Server mendeteksi bahwa *password* tidak valid dan mengembalikan pesan kesalahan.
3. Aplikasi *mobile* menampilkan pesan kesalahan dan meminta pengguna memeriksa kembali *password*.

1.4.5 Fitur Update Profil Pada Aplikasi Mobile

1.4.5.1 Deskripsi

Fitur *Update* Profil pada Aplikasi android *mobile* Maktabah YARSI memungkinkan pengguna yang telah masuk ke dalam aplikasi untuk memperbarui informasi profil mereka dan mengakses fungsi yang terkait dengan pembaruan profil.

1.4.5.2 Trigger

Fitur *Update* Profil akan diaktifkan ketika pengguna memilih opsi saya di antarmuka pengguna Aplikasi android *mobile* Maktabah YARSI.

1.4.5.3 Input

Data yang diperlukan untuk fitur ini adalah informasi akun pengguna, termasuk:

1. *Username*
2. *Alamat e-mail*
3. *Password*
4. Gambar profil

1.4.5.4 Output

Keluaran dari fitur ini adalah pembaruan informasi profil pengguna dan akses ke fungsi yang terkait dengan profil yang telah diperbarui.

1.4.5.5 Skenario Utama

Prakondisi: Aplikasi *mobile* terbuka, dan pengguna telah masuk ke dalam aplikasi.

Pascakondisi: Informasi profil pengguna diperbarui, dan pengguna mendapatkan akses penuh ke fungsi terkait dengan profil yang telah diperbarui.

Langkah-langkah:

1. Pengguna memasukkan informasi profil yang ingin diperbarui, seperti *Username*, alamat *e-mail*, *password*, dan gambar profil.
2. Aplikasi *mobile* menyimpan informasi profil yang telah diperbarui.

1.4.5.6 Skenario Eksepsional

Prakondisi: Informasi yang dimasukkan oleh pengguna seperti *Username* dan *password* kurang dari 5 karakter.

Pascakondisi: Aplikasi *mobile* memberikan pesan kesalahan dan meminta pengguna memeriksa kembali informasi profil yang dimasukkan

Langkah-langkah:

1. Aplikasi *mobile* mendeteksi *Username* dan *password* kurang dari 5 karakter.
2. Aplikasi *mobile* mengirimkan pesan kesalahan ke pengguna dan meminta pengguna memeriksa kembali informasi yang dimasukkan.
3. Pengguna memperbaiki informasi profil dan mengirimkan ulang permintaan pembaruan profil.
4. Langkah-langkah pada skenario utama dilanjutkan setelah informasi profil valid.

1.4.6 Fitur Update Profil pada Web Service

1.4.6.1 Deskripsi

Fitur *Update* Profil pada Aplikasi android *mobile* Maktabah YARSI memungkinkan pengguna yang telah masuk ke dalam aplikasi untuk memperbarui informasi profil mereka dan mengakses fungsi yang terkait dengan pembaruan profil.

1.4.6.2 *Trigger*

Fitur *Update* Profil akan diaktifkan ketika pengguna memilih opsi saya di antarmuka pengguna Aplikasi android *mobile* Maktabah YARSI.

1.4.6.3 *Input*

Data yang diperlukan untuk fitur ini adalah informasi akun pengguna, termasuk:

1. *Username*
2. *Alamat e-mail*
3. *Password*

1.4.6.4 *Output*

Keluaran dari fitur ini adalah pembaruan informasi profil pengguna dan akses ke fungsi yang terkait dengan profil yang telah diperbarui.

1.4.6.5 *Skenario Utama*

Prakondisi: Pengguna telah masuk ke dalam Aplikasi *mobile* dan berada di antarmuka pengguna yang kemudian mengklik *Username* atau *password* untuk mengubah profil.

Pascakondisi: Informasi profil pengguna diperbarui, dan pengguna mendapatkan akses penuh ke fungsi terkait dengan profil yang telah diperbarui.

Langkah-langkah:

1. Pengguna memasukkan informasi profil yang ingin diperbarui, seperti *Username, password*.
2. Layanan web mengirimkan permintaan pembaruan profil ke server.
3. Server memvalidasi informasi profil dan mengembalikan konfirmasi jika berhasil.
4. Layanan web memberikan konfirmasi kepada pengguna bahwa pembaruan profil telah berhasil dilakukan.
5. Pengguna dapat melanjutkan menggunakan layanan web dengan informasi profil yang telah diperbarui.

1.4.6.6 *Skenario Eksepsional*

Prakondisi: Informasi yang dimasukkan oleh pengguna seperti *Username* dan *password* kurang dari 5 karakter.

Pascakondisi: Layanan web memberikan pesan kesalahan dan meminta pengguna memeriksa kembali informasi profil yang dimasukkan

Langkah-langkah:

1. Layanan web mendeteksi informasi profil yang tidak valid.
2. Layanan web mengirimkan pesan kesalahan ke pengguna dan meminta pengguna memeriksa kembali informasi yang dimasukkan.
3. Pengguna memperbaiki informasi profil dan mengirimkan ulang permintaan pembaruan profil.
4. Langkah-langkah pada skenario utama dilanjutkan setelah informasi profil valid.

1.4.7 Fitur *Home* pada Aplikasi *Mobile*

1.4.7.1 Deskripsi

Fitur *Home* pada Aplikasi android *mobile* Maktabah YARSI menyediakan tampilan utama yang memuat konten dan fungsionalitas yang relevan dengan kebutuhan pengguna. Fitur ini memberikan pengguna akses cepat dan mudah ke kategori, buku terbaru, rekomendasi buku, dan tampilan *Search*.

1.4.7.2 Trigger

Fitur *Home* diaktifkan ketika pengguna membuka Aplikasi android *mobile* Maktabah YARSI atau memilih opsi "*Home*" di bottom navigasi.

1.4.7.3 Input

Tidak ada input yang perlu diisi oleh pengguna untuk menampilkan tampilan *Home*.

1.4.7.4 Output

Keluaran dari fitur ini adalah tampilan *Home* yang memiliki tampilan melihat jumlah pengunjung aplikasi, pencarian, iklan buku, kategori, buku terbaru dan buku rekomendasi.

1.4.7.5 Skenario Utama

Prakondisi: Aplikasi *mobile* terbuka atau pengguna memilih opsi "*Home*" di antarmuka pengguna.

Pascakondisi: Pengguna melihat tampilan beberapa fitur yang ada di halaman *Home*.

Langkah-langkah:

1. Pengguna memilih *Home* pada bottom navigasi
2. Aplikasi menampilkan tampilan utama

1.4.8 Fitur Iklan Buku pada Aplikasi *Mobile*

1.4.8.1 Deskripsi

Fitur iklan buku pada Aplikasi android *mobile* Maktabah YARSI menyediakan tampilan iklan buku yang memuat gambar buku yang tersedia

1.4.8.2 Trigger

Fitur iklan buku diaktifkan ketika pengguna membuka Aplikasi android *mobile* Maktabah YARSI Halaman *Home*.

1.4.8.3 Input

Tidak ada input yang perlu diisi oleh pengguna untuk menampilkan tampilan iklan buku.

1.4.8.4 Output

Keluaran dari fitur ini adalah tampilan iklan buku yang terdapat Halaman *Home*.

1.4.8.3 Skenario Utama

Prakondisi: Aplikasi *mobile* terbuka dan tampilan iklan buku berada di halaman *Home* di antarmuka pengguna.

Pascakondisi: Pengguna melihat tampilan iklan buku yang ada di halaman *Home*.

Langkah-langkah:

1. Pengguna memilih *Home* pada bottom navigasi
2. Aplikasi menampilkan iklan buku pada bagian atas di halaman *Home*

1.4.9 Fitur Kategori pada Aplikasi *Mobile*

1.4.9.1 Deskripsi

Fitur kategori pada Aplikasi android *mobile* Maktabah YARSI menyediakan tampilan kategori buku yang memuat nama kategori yang tersedia.

1.4.9.2 Trigger

Fitur iklan buku diaktifkan ketika pengguna membuka Aplikasi android *mobile* Maktabah YARSI Halaman *Home*.

1.4.9.3 Skenario Utama

Prakondisi: Aplikasi *mobile* terbuka atau pengguna berada di halaman *Home* di antarmuka pengguna.

Pascakondisi: Pengguna melihat tampilan iklan buku yang ada di halaman *Home*.

Langkah-langkah:

1. Pengguna membuka Aplikasi *mobile* dan memilih opsi "*Home*" pada antarmuka pengguna.
2. Aplikasi *mobile* menampilkan tampilan halaman *Home* dengan daftar kategori buku.
3. Pengguna melihat gambar buku yang mewakili setiap kategori yang tersedia.
4. Pengguna memilih salah satu kategori dengan mengklik gambar buku yang ingin dilihat.
5. Aplikasi *mobile* menavigasikan pengguna ke halaman khusus kategori tersebut.
6. Pengguna dapat menelusuri buku-buku dalam kategori tersebut atau memilih untuk kembali ke halaman *Home*.

1.4.10 Fitur Kategori pada *Web Service*

1.4.10.1 Deskripsi

Fitur Kategori pada *Web Service* Maktabah YARSI memberikan akses dan informasi terkini tentang kategori buku melalui layanan web atau API, termasuk gambar buku yang tersedia.

1.4.10.2 Trigger

Fitur Kategori diaktifkan ketika pengguna meminta informasi kategori buku melalui API Maktabah YARSI.

1.4.10.3 Input

Tidak ada input yang perlu diisi oleh pengguna untuk menampilkan tampilan kategori.

1.4.10.4 Output

Keluaran dari fitur ini adalah tampilan kategori yang terdapat Halaman *Home*.

1.4.10.5 Skenario Utama

Prakondisi: Pengguna melakukan permintaan informasi kategori buku melalui layanan *Web Service* atau API Maktabah YARSI.

Pascakondisi: Pengguna menerima respons yang berisi informasi kategori buku dan gambar buku yang tersedia.

Langkah-langkah:

1. Pengguna membuat permintaan GET ke API Maktabah YARSI untuk mendapatkan informasi kategori buku.
2. API Maktabah YARSI memproses permintaan dan mengambil data kategori buku dari penyimpanan data.
3. Pengguna dapat memilih salah satu kategori untuk melihat buku-buku yang termasuk dalam kategori tersebut.
4. API Maktabah YARSI menangani permintaan lebih lanjut terkait kategori buku yang dipilih.

1.4.11 Fitur Pencarian pada Aplikasi *Mobile*

1.4.11.1 Deskripsi

Fitur Pencarian pada Aplikasi android *mobile* Maktabah YARSI memungkinkan pengguna untuk mencari buku atau konten lainnya berdasarkan kata kunci tertentu, yang disediakan untuk mempercepat proses menemukan informasi yang diinginkan.

1.4.11.2 Trigger

Fitur Pencarian diaktifkan ketika pengguna memilih opsi "Pencarian" pada antarmuka pengguna Aplikasi android *mobile* Maktabah YARSI.

1.4.11.3 Input

Data yang diperlukan untuk fitur ini adalah kata kunci pencarian yang dimasukkan oleh pengguna.

1.4.11.4 Output

Keluaran dari fitur ini adalah daftar hasil pencarian yang sesuai dengan kata kunci yang dimasukkan oleh pengguna.

1.4.11.5 Skenario Utama

Prakondisi: Pengguna membuka Aplikasi *mobile* dan memilih opsi "Pencarian" pada antarmuka pengguna.

Pascakondisi: Pengguna melihat daftar hasil pencarian yang sesuai dengan kata kunci yang dimasukkan.

Langkah-langkah:

1. Pengguna memasukkan kata kunci pencarian pada antarmuka "Pencarian".
2. Aplikasi *mobile* mengirimkan permintaan pencarian ke server.
3. Server mengambil data terkait dengan kata kunci dan mengembalikan daftar hasil pencarian.
4. Aplikasi *mobile* menampilkan daftar hasil pencarian pada antarmuka pengguna.
5. Pengguna dapat memilih salah satu hasil pencarian untuk melihat detail atau melanjutkan pencarian.

1.4.12 Fitur Pencarian Kata Kunci pada Aplikasi *Mobile*

1.4.12.1 Deskripsi

Fitur Pencarian Kata Kunci pada Aplikasi android *mobile* Maktabah YARSI memungkinkan pengguna untuk mencari buku atau konten lainnya berdasarkan kata kunci tertentu, yang memberikan pengalaman pencarian yang lebih terfokus dan relevan.

1.4.12.2 Trigger

Fitur Pencarian Kata Kunci diaktifkan ketika pengguna memilih opsi "Pencarian" dan memasukkan kata kunci tertentu pada antarmuka pengguna Aplikasi android *mobile* Maktabah YARSI.

1.4.12.3 Input

Data yang diperlukan untuk fitur ini adalah kata kunci pencarian yang dimasukkan oleh pengguna.

1.4.12.4 Output

Keluaran dari fitur ini adalah daftar hasil pencarian yang sesuai dengan kata kunci yang dimasukkan oleh pengguna.

1.4.12.5 Skenario Utama

Prakondisi: Pengguna membuka Aplikasi *mobile* dan memilih opsi "Pencarian" pada antarmuka pengguna.

Pascakondisi: Pengguna melihat daftar hasil pencarian yang sesuai dengan kata kunci yang dimasukkan.

Langkah-langkah:

1. Pengguna memilih opsi "Pencarian" pada antarmuka pengguna.
2. Aplikasi *mobile* menampilkan kotak teks untuk memasukkan kata kunci pencarian.
3. Pengguna memasukkan kata kunci yang diinginkan pada kotak teks.
4. Pengguna menekan tombol "Cari" atau setara untuk memulai proses pencarian.
5. Aplikasi *mobile* mengirimkan permintaan pencarian ke server dengan kata kunci yang dimasukkan.
6. Server mengambil data terkait dengan kata kunci dan mengembalikan daftar hasil pencarian.
7. Aplikasi *mobile* menampilkan daftar hasil pencarian pada antarmuka pengguna.
8. Pengguna dapat memilih salah satu hasil pencarian untuk melihat detail atau melanjutkan pencarian.

1.4.12.6 Skenario Eksepsional

Prakondisi: Tidak ada hasil pencarian yang sesuai dengan kata kunci yang dimasukkan.

Pascakondisi: Aplikasi *mobile* menampilkan pesan informasi bahwa tidak ada hasil pencarian yang ditemukan.

Langkah-langkah:

1. Aplikasi *mobile* mendeteksi bahwa tidak ada hasil pencarian yang sesuai dengan kata kunci.
2. Aplikasi *mobile* menampilkan pesan bahwa hasil pencarian tidak ditemukan.
3. Pengguna dapat memodifikasi kata kunci atau melakukan tindakan lainnya pada antarmuka "Pencarian".

1.4.13 Fitur Pencarian Kata Kunci pada Web Service

1.4.13.1 Deskripsi

Fitur Pencarian Kata Kunci pada *Web Service* Maktabah YARSI memungkinkan pengguna untuk mencari buku atau konten lainnya berdasarkan kata kunci tertentu

melalui layanan web atau API, yang memberikan pengalaman pencarian yang lebih terfokus dan relevan.

1.4.13.2 Trigger

Fitur Pencarian Kata Kunci diaktifkan ketika pengguna meminta informasi pencarian dengan kata kunci tertentu melalui layanan web Maktabah YARSI.

1.4.13.3 Input

Data yang diperlukan untuk fitur ini adalah kata kunci pencarian yang dimasukkan oleh pengguna.

1.4.13.4 Output

Keluaran dari fitur ini adalah daftar hasil pencarian yang sesuai dengan kata kunci yang dimasukkan oleh pengguna.

1.4.13.5 Skenario Utama

Prakondisi: Pengguna melakukan permintaan informasi pencarian dengan kata kunci tertentu melalui layanan web atau API Maktabah YARSI.

Pascakondisi: Pengguna menerima respons yang berisi daftar hasil pencarian yang sesuai dengan kata kunci yang dimasukkan.

Langkah-langkah:

1. Pengguna membuat permintaan pencarian ke API Maktabah YARSI dengan menyertakan kata kunci.
2. API Maktabah YARSI memproses permintaan dan mengambil data terkait dengan kata kunci dari penyimpanan data.
3. API mengembalikan respons yang berisi daftar hasil pencarian yang sesuai dengan kata kunci.
4. Pengguna menerima respons dan menampilkan daftar hasil pencarian pada aplikasinya atau situs web.
5. Pengguna dapat memilih salah satu hasil pencarian untuk melihat detail atau melanjutkan pencarian.

1.4.13.6 Skenario Eksepsional

Prakondisi: Tidak ada hasil pencarian yang sesuai dengan kata kunci yang dimasukkan.

Pascakondisi: API Maktabah YARSI memberikan respons yang menyatakan bahwa tidak ada hasil pencarian yang ditemukan.

Langkah-langkah:

1. API Maktabah YARSI mendeteksi bahwa tidak ada hasil pencarian yang sesuai dengan kata kunci.
2. API mengembalikan respons yang berisi pesan informasi bahwa tidak ada hasil pencarian yang tersedia saat ini.
3. Pengguna menerima respons dan dapat menangani pesan informasi tersebut pada aplikasinya atau situs web.

1.4.14 Fitur *Highlight* Pencarian pada Aplikasi *Mobile*

1.4.14.1 Deskripsi

Fitur *Highlight* Pencarian pada Aplikasi android *mobile* Maktabah YARSI memungkinkan pengguna untuk menyorot atau membedakan *list* judul dan isi buku yang sesuai dengan kata kunci pencarian, memudahkan pengguna untuk melihat informasi yang relevan.

1.4.14.2 Trigger

Fitur *Highlight* Pencarian diaktifkan ketika pengguna melakukan pencarian kata kunci dan hasil pencarian ditampilkan pada antarmuka pengguna Aplikasi android *mobile* Maktabah YARSI.

1.4.14.3 Input

Data yang diperlukan untuk fitur ini adalah kata kunci pencarian yang dimasukkan oleh pengguna dan data buku yang sesuai dengan pencarian tersebut.

1.4.14.4 Output

Keluaran dari fitur ini adalah daftar hasil pencarian dengan judul dan isi buku yang telah di-*highlight* berdasarkan kata kunci pencarian.

1.4.14.5 Skenario Utama

Prakondisi: Pengguna telah melakukan pencarian dengan kata kunci tertentu dan melihat hasil pencarian pada antarmuka pengguna Aplikasi *mobile*.

Pascakondisi: Pengguna melihat daftar hasil pencarian dengan judul dan isi buku yang di-*highlight* berdasarkan kata kunci pencarian.

Langkah-langkah:

1. Pengguna melakukan pencarian dengan memasukkan kata kunci pada antarmuka "Pencarian".
2. Aplikasi *mobile* mengirimkan permintaan pencarian ke server.
3. Server mengambil data terkait dengan kata kunci dan mengembalikan daftar hasil pencarian.
4. Aplikasi *mobile* menampilkan daftar hasil pencarian dengan judul dan isi buku.
5. Bagian judul dan isi buku yang sesuai dengan kata kunci pencarian di-*highlight* atau diberi tanda khusus.
6. Pengguna dapat dengan mudah melihat bagian-bagian yang relevan dengan pencarian mereka.

1.4.14.6 Skenario Eksepsional

Prakondisi: Tidak ada hasil pencarian yang sesuai dengan kata kunci yang dimasukkan.

Pascakondisi: Aplikasi *mobile* menampilkan pesan informasi bahwa tidak ada hasil pencarian yang ditemukan.

Langkah-langkah:

1. Aplikasi *mobile* mendeteksi bahwa tidak ada hasil pencarian yang sesuai dengan kata kunci.
2. Aplikasi *mobile* menampilkan pesan informasi kepada pengguna.
3. Pengguna dapat memodifikasi kata kunci atau melakukan tindakan lainnya pada antarmuka "Pencarian".

1.4.15 Fitur *Highlight* Pencarian pada *Web Service*

1.4.15.1 Deskripsi

Fitur *Highlight* Pencarian pada *Web Service* Maktabah YARSI memungkinkan pengguna untuk menyorot atau membedakan *list* judul dan isi buku yang sesuai dengan kata kunci pencarian, memberikan pengalaman yang lebih terfokus dan relevan melalui layanan web atau API.

1.4.15.2 Trigger

Fitur *Highlight* Pencarian diaktifkan ketika pengguna melakukan permintaan pencarian kata kunci melalui layanan web atau API Maktabah YARSI.

1.4.15.3 Input

Data yang diperlukan untuk fitur ini adalah kata kunci pencarian yang dimasukkan oleh pengguna dan data buku yang sesuai dengan pencarian tersebut.

1.4.15.4 Output

Keluaran dari fitur ini adalah daftar hasil pencarian dengan judul dan isi buku yang telah di-*highlight* berdasarkan kata kunci pencarian.

1.4.15.5 Skenario Utama

Prakondisi: Pengguna melakukan permintaan pencarian dengan kata kunci tertentu melalui layanan web atau API Maktabah YARSI.

Pascakondisi: Pengguna menerima respons yang berisi daftar hasil pencarian dengan judul dan isi buku yang di-*highlight* berdasarkan kata kunci pencarian.

Langkah-langkah:

1. Pengguna membuat permintaan pencarian ke layanan web atau API Maktabah YARSI dengan menyertakan kata kunci.
2. Layanan web atau API Maktabah YARSI memproses permintaan dan mengambil data terkait dengan kata kunci dari penyimpanan data.
3. Layanan web atau API mengembalikan respons yang berisi daftar hasil pencarian dengan judul dan isi buku.
4. Bagian judul dan isi buku yang sesuai dengan kata kunci pencarian di-*highlight* atau diberi tanda khusus dalam respons.
5. Pengguna menerima respons dan dapat dengan mudah melihat bagian-bagian yang relevan dengan pencarian mereka.

1.4.15.6 Skenario Eksepsional

Prakondisi: Tidak ada hasil pencarian yang sesuai dengan kata kunci yang dimasukkan.

Pascakondisi: Layanan web atau API Maktabah YARSI memberikan respons yang menyatakan bahwa tidak ada hasil pencarian yang ditemukan.

Langkah-langkah:

1. Layanan web atau API Maktabah YARSI mendeteksi bahwa tidak ada hasil pencarian yang sesuai dengan kata kunci.
2. Layanan web atau API mengembalikan respons yang berisi pesan informasi bahwa tidak ada hasil pencarian yang tersedia saat ini.
3. Pengguna menerima respons dan dapat menangani pesan informasi tersebut pada aplikasinya atau situs web.

1.4.16 Fitur Detail Buku pada Aplikasi *Mobile*

1.4.16.1 Deskripsi

Fitur Detail Buku pada Aplikasi android *mobile* Maktabah YARSI memungkinkan pengguna untuk melihat informasi lengkap mengenai sebuah buku setelah mereka memilih atau mengklik buku tersebut dari hasil pencarian atau kategori.

1.4.16.2 Trigger

Fitur Detail Buku diaktifkan ketika pengguna memilih atau mengklik buku tertentu dari hasil pencarian atau kategori pada antarmuka pengguna Aplikasi android *mobile* Maktabah YARSI.

1.4.16.3 Input

Data yang diperlukan untuk fitur ini adalah identifikasi unik atau data buku yang dipilih oleh pengguna.

1.4.16.4 Output

Keluaran dari fitur ini adalah tampilan halaman detail buku yang berisi informasi lengkap tentang buku yang dipilih, seperti judul, pengarang, jumlah buku terbaca, dan tahun terbit.

1.4.16.5 Skenario Utama

Prakondisi: Pengguna telah melakukan pencarian atau memilih kategori dan melihat hasil pencarian atau daftar kategori buku pada antarmuka pengguna Aplikasi *mobile*.

Pascakondisi: Pengguna melihat halaman detail buku setelah memilih atau mengklik buku tertentu dari hasil pencarian atau kategori.

Langkah-langkah:

1. Pengguna melakukan pencarian atau memilih kategori pada antarmuka pengguna.

2. Aplikasi *mobile* menampilkan daftar hasil pencarian atau kategori buku.
3. Pengguna memilih atau mengklik salah satu buku dari daftar hasil.
4. Aplikasi *mobile* menavigasikan pengguna ke halaman detail buku yang dipilih.
5. Pengguna melihat informasi lengkap mengenai buku, seperti judul, pengarang, sinopsis, dan lainnya.
6. Pengguna dapat melakukan aksi seperti meminjam buku, menambahkan ke daftar favorit, atau kembali ke halaman sebelumnya.

1.4.16.6 Skenario Eksepsional

Prakondisi: Buku yang dipilih tidak memiliki informasi detail yang cukup.

Pascakondisi: Aplikasi *mobile* menampilkan pesan informasi bahwa detail buku tidak tersedia saat ini.

Langkah-langkah:

1. Aplikasi *mobile* mendeteksi bahwa buku yang dipilih tidak memiliki informasi detail yang cukup.
2. Aplikasi *mobile* menampilkan tampilan kosong kepada pengguna.
3. Pengguna dapat kembali ke halaman sebelumnya atau melakukan tindakan lainnya pada antarmuka aplikasi.

1.4.17 Fitur Detail Buku pada Web Service

1.4.17.1 Deskripsi

Fitur Detail Buku pada *Web Service* Maktabah YARSI memungkinkan pengguna untuk mendapatkan informasi lengkap mengenai sebuah buku melalui layanan web atau API setelah melakukan permintaan terhadap buku tertentu.

1.4.17.2 Trigger

Fitur Detail Buku diaktifkan ketika pengguna melakukan permintaan detail buku melalui layanan web atau API Maktabah YARSI dengan menyertakan identifikasi unik atau data buku yang diinginkan.

1.4.17.3 Input

Data yang diperlukan untuk fitur ini adalah identifikasi unik atau data buku yang diminta oleh pengguna.

1.4.17.4 Output

Keluaran dari fitur ini adalah respons berisi informasi lengkap mengenai buku yang diminta, seperti judul, pengarang, sinopsis, dan detail lainnya.

1.4.17.5 Skenario Utama

Prakondisi: Pengguna melakukan permintaan detail buku dengan menyertakan identifikasi unik melalui layanan web atau API Maktabah YARSI.

Pascakondisi: Pengguna menerima respons berisi informasi lengkap mengenai buku yang diminta.

Langkah-langkah:

1. Pengguna membuat permintaan detail buku ke layanan web atau API Maktabah YARSI dengan menyertakan identifikasi unik atau data buku yang diminta.
2. Layanan web atau API memproses permintaan dan mengambil data terkait buku dari penyimpanan data.
3. Layanan web atau API mengembalikan respons yang berisi informasi lengkap tentang buku yang diminta.
4. Pengguna menerima respons dan dapat menampilkan informasi lengkap mengenai buku pada aplikasinya atau situs web.

1.4.17.6 Skenario Eksepsional

Prakondisi: Informasi detail buku tidak tersedia pada layanan web atau API.

Pascakondisi: Layanan web atau API memberikan respons yang menyatakan bahwa informasi detail buku tidak tersedia saat ini.

Langkah-langkah:

1. Layanan web atau API mendeteksi bahwa informasi detail buku tidak tersedia.
2. Layanan web atau API mengembalikan respons yang berisi pesan informasi bahwa detail buku tidak tersedia saat ini.
3. Pengguna menerima respons dan dapat menangani pesan informasi tersebut pada aplikasinya atau situs web.

1.4.18 Fitur Analisis Jumlah Buku pada Aplikasi *Mobile*

1.4.18.1 Deskripsi

Fitur Analisis Jumlah Buku pada Aplikasi android *mobile* Maktabah YARSI memungkinkan pengguna untuk melihat statistik atau analisis terkait jumlah buku yang terdapat di halaman *Home*, khususnya yang terletak di dekat kategori buku.

1.4.18.2 Trigger

Fitur Analisis Jumlah Buku diaktifkan ketika pengguna memilih atau mengklik bagian yang menyajikan analisis jumlah buku Halaman *Home* Aplikasi android *mobile* Maktabah YARSI.

1.4.18.3 Input

Data yang diperlukan untuk fitur ini adalah identifikasi atau kategori buku yang diinginkan oleh pengguna untuk dianalisis jumlahnya.

1.4.18.4 Output

Keluaran dari fitur ini adalah tampilan grafik atau statistik yang menunjukkan jumlah buku yang terdapat di dekat kategori tertentu Halaman *Home*.

1.4.18.5 Skenario Utama

Prakondisi: Pengguna membuka Aplikasi *mobile* dan berada di halaman *Home*.

Pascakondisi: Pengguna melihat analisis jumlah buku yang terdapat di dekat kategori tertentu Halaman *Home*.

Langkah-langkah:

1. Pengguna melihat bagian atau opsi yang menyajikan analisis jumlah buku di dekat kategori Halaman *Home*.
2. Pengguna memilih atau mengklik bagian tersebut untuk melihat analisis lebih lanjut.
3. Aplikasi *mobile* menampilkan grafik atau statistik yang menunjukkan jumlah buku yang terdapat di dekat kategori tersebut.
4. Pengguna dapat melihat informasi seperti total buku, rasio jenis buku, atau informasi lainnya terkait jumlah buku di dekat kategori tersebut.

1.4.18.6 Skenario Eksepsional

Prakondisi: Tidak ada data atau informasi untuk dilakukan analisis jumlah buku.

Pascakondisi: Aplikasi *mobile* menampilkan pesan informasi bahwa tidak ada data atau informasi yang tersedia untuk dilakukan analisis jumlah buku.

Langkah-langkah:

1. Aplikasi *mobile* mendeteksi bahwa tidak ada data atau informasi untuk dilakukan analisis jumlah buku.
2. Aplikasi *mobile* menampilkan pesan informasi kepada pengguna.
3. Pengguna dapat kembali ke halaman *Home* atau melakukan tindakan lainnya pada antarmuka aplikasi.

1.4.19 Fitur Analisis Jumlah Buku pada Web Service

1.4.19.1 Deskripsi

Fitur Analisis Jumlah Buku pada *Web Service* Maktabah YARSI memungkinkan pengguna untuk mendapatkan informasi statistik atau analisis terkait jumlah buku yang terdapat di dekat kategori melalui layanan web atau API.

1.4.19.2 Trigger

Fitur Analisis Jumlah Buku diaktifkan ketika pengguna melakukan permintaan analisis jumlah buku melalui layanan web atau API Maktabah YARSI dengan menyertakan identifikasi atau kategori buku yang diinginkan.

1.4.19.3 Input

Data yang diperlukan untuk fitur ini adalah identifikasi atau kategori buku yang diinginkan oleh pengguna untuk dianalisis jumlahnya.

1.4.19.4 Output

Keluaran dari fitur ini adalah respons yang berisi informasi statistik atau analisis jumlah buku yang terdapat di kategori tertentu.

1.4.19.5 Skenario Utama

Prakondisi: Pengguna melakukan permintaan analisis jumlah buku dengan menyertakan identifikasi atau kategori buku melalui layanan web atau API Maktabah YARSI.

Pascakondisi: Pengguna menerima respons berisi informasi statistik atau analisis jumlah buku yang terdapat di kategori tertentu.

Langkah-langkah:

1. Pengguna membuat permintaan analisis jumlah buku ke layanan web atau API Maktabah YARSI dengan menyertakan identifikasi atau kategori buku yang diinginkan.
2. Layanan web atau API memproses permintaan dan mengambil data terkait jumlah buku di dekat kategori tersebut dari penyimpanan data.
3. Layanan web atau API mengembalikan respons yang berisi informasi statistik atau analisis jumlah buku yang terdapat di dekat kategori tertentu.
4. Pengguna menerima respons dan dapat menampilkan informasi statistik pada aplikasinya atau situs web.

1.4.19.6 Skenario Eksepsional

Prakondisi: Tidak ada data atau informasi untuk dilakukan analisis jumlah buku.

Pascakondisi: Layanan web atau API Maktabah YARSI memberikan respons yang menyatakan bahwa tidak ada data atau informasi yang tersedia untuk dilakukan analisis jumlah buku.

Langkah-langkah:

1. Layanan web atau API mendeteksi bahwa tidak ada data atau informasi untuk dilakukan analisis jumlah buku.
2. Layanan web atau API mengembalikan respons yang berisi pesan informasi bahwa tidak ada data atau informasi yang tersedia saat ini.
3. Pengguna menerima respons dan dapat menangani pesan informasi tersebut pada aplikasinya atau situs web.

1.4.20 Fitur Membaca Isi Buku pada Aplikasi *Mobile*

1.4.20.1 Deskripsi

Fitur Membaca Isi Buku pada Aplikasi android *mobile* Maktabah YARSI memungkinkan pengguna untuk membaca isi lengkap buku setelah mereka memilih atau mengklik buku

tertentu dari halaman detail buku. Pengguna dapat mulai membaca konten buku langsung dari aplikasi.

1.4.20.2 Trigger

Fitur Membaca Isi Buku diaktifkan ketika pengguna memilih atau mengklik buku tertentu dari halaman detail buku dan memilih opsi "Mulai Membaca".

1.4.20.3 Input

Data yang diperlukan untuk fitur ini adalah identifikasi unik atau data buku yang dipilih oleh pengguna.

1.4.20.4 Output

Keluaran dari fitur ini adalah tampilan halaman membaca yang memuat isi lengkap buku, termasuk fitur-fitur seperti penanda halaman, penyesuaian ukuran teks, dan lainnya.

1.4.20.5 Skenario Utama

Prakondisi: Pengguna telah memilih atau mengklik buku tertentu dari halaman detail buku pada antarmuka pengguna Aplikasi *mobile*.

Pascakondisi: Pengguna dapat membaca isi lengkap buku setelah memilih opsi "Mulai Membaca".

Langkah-langkah:

1. Pengguna memilih atau mengklik buku tertentu dari halaman detail buku.
2. Aplikasi *mobile* menavigasikan pengguna ke halaman membaca dan memuat isi lengkap buku.
3. Pengguna dapat menggunakan fitur-fitur seperti penanda halaman, penyesuaian ukuran teks, atau fitur membaca lainnya yang disediakan oleh aplikasi.
4. Pengguna dapat membalik halaman, mencari halaman tertentu, atau kembali ke halaman sebelumnya.

1.4.20.6 Skenario Eksepsional

Prakondisi: Buku yang dipilih tidak memiliki isi yang dapat dibaca.

Pascakondisi: Aplikasi *mobile* menampilkan pesan informasi bahwa isi buku tidak tersedia saat ini.

Langkah-langkah:

1. Aplikasi *mobile* mendeteksi bahwa buku yang dipilih tidak memiliki isi yang dapat dibaca.
2. Aplikasi *mobile* menampilkan pesan informasi kepada pengguna.
3. Pengguna diberikan opsi untuk kembali ke halaman sebelumnya atau melakukan tindakan lainnya pada antarmuka aplikasi.

1.4.21 Fitur Membaca Isi Buku pada *Web Service*

1.4.21.1 Deskripsi

Fitur Membaca Isi Buku pada *Web Service* Maktabah YARSI memungkinkan pengguna untuk membaca isi lengkap buku setelah mereka melakukan permintaan terhadap buku tertentu melalui layanan *Web Service* atau API. Pengguna dapat mulai membaca konten buku langsung dari aplikasi.

1.4.21.2 Trigger

Fitur Membaca Isi Buku diaktifkan ketika pengguna melakukan permintaan membaca isi buku melalui layanan web atau API Maktabah YARSI dengan menyertakan identifikasi unik atau data buku yang diinginkan.

1.4.21.3 Input

Data yang diperlukan untuk fitur ini adalah identifikasi unik atau data buku yang diminta oleh pengguna.

1.4.21.4 Output

Keluaran dari fitur ini adalah respons yang berisi tampilan halaman membaca yang memuat isi lengkap buku, termasuk fitur-fitur seperti penanda halaman, penyesuaian ukuran teks, dan lainnya.

1.4.21.5 Skenario Utama

Prakondisi: Pengguna melakukan permintaan membaca isi buku dengan menyertakan identifikasi unik atau data buku melalui layanan web atau API Maktabah YARSI.

Pascakondisi: Pengguna menerima respons berisi tampilan halaman membaca yang memuat isi lengkap buku.

Langkah-langkah:

1. Pengguna membuat permintaan membaca isi buku ke layanan web atau API Maktabah YARSI dengan menyertakan identifikasi unik atau data buku yang diminta.
2. Layanan web atau API memproses permintaan dan mengambil data terkait isi buku dari penyimpanan data.
3. Layanan web atau API mengembalikan respons yang berisi tampilan halaman membaca yang memuat isi lengkap buku.
4. Pengguna dapat menggunakan fitur-fitur seperti penanda halaman, penyesuaian ukuran teks, atau fitur membaca lainnya yang disediakan oleh aplikasi atau situs web yang mengintegrasikan layanan tersebut.

1.4.21.6 Skenario Eksepsional

Prakondisi: Buku yang diminta tidak memiliki isi yang dapat dibaca.

Pascakondisi: Layanan web atau API Maktabah YARSI memberikan respons yang menyatakan bahwa isi buku tidak tersedia saat ini.

Langkah-langkah:

1. Layanan web atau API mendeteksi bahwa buku yang diminta tidak memiliki isi yang dapat dibaca.
2. Layanan web atau API mengembalikan respons yang berisi pesan informasi bahwa isi buku tidak tersedia saat ini.
3. Pengguna menerima respons dan dapat menangani pesan informasi tersebut pada aplikasinya atau situs web.

1.4.22 Fitur Menambahkan Buku ke Favorit pada Aplikasi *Mobile*

1.4.22.1 Deskripsi

Fitur Menambahkan Buku ke Favorit pada Aplikasi android *mobile* Maktabah YARSI memungkinkan pengguna untuk menyimpan buku tertentu ke dalam daftar buku favorit mereka untuk referensi atau akses lebih cepat di masa mendatang. Pengguna dapat melakukan tindakan ini melalui halaman detail buku.

1.4.22.2 *Trigger*

Fitur ini diaktifkan ketika pengguna memilih atau mengklik bintang favorit Halaman detail buku pada antarmuka pengguna Aplikasi android *mobile* Maktabah YARSI.

1.4.22.3 Input

Data yang diperlukan untuk fitur ini adalah data buku yang ingin ditambahkan ke daftar buku favorit pengguna.

1.4.22.4 Output

Keluaran dari fitur ini adalah konfirmasi bahwa buku telah berhasil ditambahkan ke dalam daftar buku favorit pengguna.

1.4.22.5 Skenario Utama

Prakondisi: Pengguna membuka Aplikasi *mobile* dan berada di halaman detail buku.

Pascakondisi: Pengguna menerima konfirmasi bahwa buku telah ditambahkan ke dalam daftar buku favorit mereka.

Langkah-langkah:

1. Pengguna memilih atau mengklik buku tertentu dari halaman detail buku.
2. Aplikasi *mobile* menampilkan informasi lengkap mengenai buku tersebut.
3. Pengguna mengklik ikon bintang yang terdapat Halaman detail buku.
4. Aplikasi *mobile* menyimpan data buku ke dalam daftar buku favorit pengguna.
5. Pengguna menerima konfirmasi bahwa buku telah ditambahkan ke dalam daftar buku favorit mereka.

1.4.22.6 Skenario Eksepsional

Prakondisi: Buku yang dipilih tidak dapat ditambahkan ke dalam daftar buku favorit.

Pascakondisi: Aplikasi *mobile* menampilkan pesan informasi bahwa buku tidak dapat ditambahkan ke dalam daftar buku favorit.

Langkah-langkah:

1. Aplikasi *mobile* mendeteksi bahwa buku yang dipilih tidak dapat ditambahkan ke dalam daftar buku favorit.
2. Aplikasi *mobile* menampilkan pesan informasi kepada pengguna.

3. Pengguna diberikan opsi untuk mencoba lagi atau melakukan tindakan lainnya pada antarmuka aplikasi.

1.4.23 Fitur Melihat Riwayat Baca pada Aplikasi *Mobile*

1.4.23.1 Deskripsi

Fitur Melihat Riwayat Baca pada Aplikasi android *mobile* Maktabah YARSI memungkinkan pengguna untuk melihat daftar buku yang terakhir mereka baca atau akses. Fitur ini mempermudah pengguna untuk kembali ke buku yang pernah dibaca sebelumnya.

1.4.23.2 Trigger

Fitur ini diaktifkan ketika pengguna memilih atau mengklik opsi "Riwayat" pada *bottom navigasi* antarmuka pengguna Aplikasi android *mobile* Maktabah YARSI.

1.4.23.3 Input

Tidak ada *input* yang diperlukan untuk fitur ini, karena daftar riwayat baca dihasilkan berdasarkan data buku yang telah diakses oleh pengguna sebelumnya.

1.4.23.4 Output

Keluaran dari fitur ini adalah tampilan daftar buku yang terakhir kali diakses oleh pengguna, lengkap dengan informasi buku.

1.4.23.5 Skenario Utama

Prakondisi: Pengguna membuka Aplikasi *mobile* dan berada di halaman utama atau *Home*.

Pascakondisi: Pengguna melihat daftar buku yang terakhir kali diakses pada fitur "Riwayat".

Langkah-langkah:

1. Pengguna memilih atau mengklik opsi "Riwayat Baca" pada antarmuka pengguna.
2. Aplikasi *mobile* menampilkan daftar buku yang terakhir kali diakses oleh pengguna, termasuk informasi buku.

3. Pengguna dapat memilih salah satu buku dari daftar untuk membuka halaman detail buku atau memulai membaca kembali.

1.4.23.6 Skenario Eksepsional

Prakondisi: Riwayat baca pengguna kosong atau belum ada data.

Pascakondisi: Aplikasi *mobile* menampilkan pesan informasi bahwa riwayat baca pengguna masih kosong.

Langkah-langkah:

1. Aplikasi *mobile* mendeteksi bahwa riwayat baca pengguna masih kosong atau belum ada data.
2. Aplikasi *mobile* menampilkan pesan informasi kepada pengguna.
3. Pengguna dapat memulai membaca buku baru atau melakukan tindakan lainnya pada antarmuka aplikasi.

1.4.24 Fitur Ganti Tema pada Aplikasi *Mobile*

1.4.24.1 Deskripsi

Fitur Ganti Tema pada Aplikasi android *mobile* Maktabah YARSI memungkinkan pengguna untuk beralih antara tema gelap dan terang dengan menggunakan tombol *switch*. Fitur ini dapat diakses melalui halaman akun pengguna.

1.4.24.2 Trigger

Fitur ini diaktifkan ketika pengguna menggeser atau menekan tombol *switch* "Ganti Tema" Halaman akun pengguna Aplikasi android *mobile* Maktabah YARSI.

1.4.24.3 Input

Data yang diperlukan untuk fitur ini adalah status *switch* (aktif/nonaktif) yang menunjukkan tema yang dipilih oleh pengguna (gelap/terang).

1.4.24.4 Output

Keluaran dari fitur ini adalah perubahan tema tampilan pada seluruh aplikasi sesuai dengan status *switch* yang dipilih oleh pengguna.

1.4.24.5 Skenario Utama

Prakondisi: Pengguna membuka Aplikasi *mobile* dan berada di halaman akun pengguna.

Pascakondisi: Pengguna melihat perubahan tema tampilan pada seluruh aplikasi setelah menggeser atau menekan *switch* "Ganti Tema".

Langkah-langkah:

1. Pengguna menggeser atau menekan *switch* "Ganti Tema" Halaman akun pengguna.
2. Aplikasi *mobile* menanggapi perubahan status *switch* dan memutuskan tema yang akan diaktifkan berdasarkan status tersebut.
3. Aplikasi *mobile* menerapkan perubahan tema tampilan sesuai dengan tema yang dipilih oleh pengguna.
4. Pengguna melihat perubahan tema tampilan pada seluruh aplikasi.

1.4.25 Fitur Menampilkan Jumlah Pengunjung Pada Aplikasi *mobile*

1.4.25.1 Deskripsi

Fitur Menampilkan Jumlah Pengunjung pada Aplikasi android *mobile* Maktabah YARSI memungkinkan pengguna untuk melihat informasi tentang jumlah pengunjung atau pengguna aktif pada aplikasi. Informasi ini dapat diakses melalui ikon mata dan jumlah angka yang ditampilkan Halaman *Home*.

1.4.25.2 Trigger

Fitur ini diaktifkan ketika pengguna sudah masuk ke dalam Halaman *Home* lalu akan menampilkan informasi jumlah pengunjung Halaman *Home* Aplikasi android *mobile* Maktabah YARSI.

1.4.25.3 Input

Tidak ada *input* yang diperlukan untuk fitur ini karena informasi jumlah pengunjung diambil secara otomatis berdasarkan data aktifitas pengguna pada aplikasi.

1.4.25.4 Output

Keluaran dari fitur ini adalah tampilan informasi jumlah pengunjung pada aplikasi yang mencakup periode waktu tertentu.

1.4.25.5 Skenario Utama

Prakondisi: Pengguna membuka Aplikasi *mobile* dan berada di halaman *Home*.

Pascakondisi: Pengguna melihat informasi jumlah pengunjung setelah masuk ke dalam Halaman *Home*.

Langkah-langkah:

1. Pengguna masuk ke dalam aplikasi lalu akan menampilkan informasi jumlah pengunjung Halaman *Home*.
2. Aplikasi *mobile* menampilkan informasi jumlah pengunjung yang mencakup periode waktu tertentu.
3. Pengguna dapat melihat atau menutup informasi jumlah pengunjung tersebut.

1.4.26 Fitur Menampilkan Jumlah Pengunjung pada Web Service

1.4.26.1 Deskripsi

Fitur Menampilkan Jumlah Pengunjung pada Web Service Maktabah YARSI memungkinkan pengguna untuk melihat informasi tentang jumlah pengunjung atau pengguna aktif pada aplikasi. Informasi ini dapat diakses melalui panggilan API khusus yang menyediakan data jumlah pengunjung.

1.4.26.2 Trigger

Fitur ini diaktifkan ketika pengguna membuat panggilan API khusus yang disediakan oleh Web Service Maktabah YARSI untuk mendapatkan informasi jumlah pengunjung.

1.4.26.3 Input

Data yang diperlukan untuk fitur ini adalah parameter atau token otentikasi yang memungkinkan pengguna mengakses data jumlah pengunjung melalui panggilan API.

1.4.26.4 Output

Keluaran dari fitur ini adalah respons API yang berisi informasi jumlah pengunjung pada Aplikasi android *mobile* Maktabah YARSI.

1.4.26.5 Skenario Utama

Prakondisi: Pengguna membuat panggilan API khusus dengan menyertakan parameter atau token otentikasi.

Pascakondisi: Pengguna menerima respons API yang berisi informasi jumlah pengunjung.

Langkah-langkah:

1. Pengguna membuat panggilan API khusus dengan menyertakan parameter atau token otentikasi.
2. *Web Service* Maktabah YARSI memproses panggilan API dan mengambil data jumlah pengunjung dari penyimpanan data.
3. *Web Service* Maktabah YARSI mengembalikan respons API yang berisi informasi jumlah pengunjung.
4. Pengguna dapat menggunakan data tersebut untuk menampilkan informasi jumlah pengunjung pada aplikasi.

1.4.27 Fitur Keluar pada Aplikasi *Mobile*

1.4.27.1 Deskripsi

Fitur Keluar pada Aplikasi android *mobile* Maktabah YARSI memungkinkan pengguna untuk keluar dari akun mereka dan mengakhiri sesi aktif. Tombol keluar ini terletak Halaman akun pengguna.

1.4.27.2 Trigger

Fitur ini diaktifkan ketika pengguna memilih atau mengklik tombol “Keluar” Halaman akun pengguna Aplikasi android *mobile* Maktabah YARSI.

1.4.27.3 Input

Tidak ada *input* yang diperlukan untuk fitur ini, karena fitur ini hanya memerlukan tindakan pengguna untuk mengklik tombol "Keluar".

1.4.27.4 Output

Keluaran dari fitur ini adalah keluar dari akun pengguna dan kembali ke halaman Masuk atau halaman awal Aplikasi android *mobile* Maktabah YARSI.

1.4.27.5 Skenario Utama

Prakondisi: Pengguna sudah masuk ke dalam akun pada Aplikasi android *mobile* Maktabah YARSI dan berada di halaman akun.

Pascakondisi: Pengguna keluar dari akun dan diarahkan kembali ke halaman Masuk atau halaman awal Aplikasi *mobile*.

Langkah-langkah:

1. Pengguna memilih atau mengklik tombol "Keluar" Halaman akun pengguna.
2. Aplikasi *mobile* mengakhiri sesi pengguna dan mengarahkannya kembali ke halaman Masuk atau halaman awal.
3. Pengguna melihat konfirmasi bahwa mereka telah berhasil keluar dari akun.

1.4.28 Fitur Pengelola Data Menambah Kategori Buku

1.4.28.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk menambahkan kategori buku baru ke dalam sistem. Kategori buku ini memerlukan beberapa atribut, termasuk *_id* sebagai *primary key*, *name* sebagai nama kategori, dan *total* sebagai total buku dalam kategori.

1.4.28.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan POST melalui *Postman* atau alat pengelola API lainnya untuk menambahkan kategori buku baru.

1.4.28.3 Input

Data yang diperlukan untuk menambah kategori buku melibatkan parameter sebagai berikut:

1. *_id* (objectId) [*Primary key*]: ID unik sebagai kunci utama.
2. *name* (string): Nama kategori buku.
3. *total* (number): Total buku dalam kategori.

1.4.28.4 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang memberikan konfirmasi penambahan kategori buku atau memberikan pesan kesalahan jika ada.

1.4.28.5 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang diperlukan.

Pascakondisi: Kategori buku baru berhasil ditambahkan ke dalam sistem.

Langkah-langkah:

1. Pengguna membuat permintaan POST ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter yang dibutuhkan, seperti *_id* (*objectId*), *name*, dan *total*.
3. *Web Service* Maktabah YARSI memproses permintaan dan menambahkan kategori buku baru ke dalam sistem.
4. *Web Service* Maktabah YARSI mengirim respons yang berisi konfirmasi penambahan kategori buku atau pesan kesalahan jika ada.
5. Pengguna melihat respons dari *Postman* yang menunjukkan status penambahan kategori buku.

1.4.28.6 Skenario Eksepsional

Prakondisi: Terjadi kesalahan dalam proses penambahan kategori buku.

Pascakondisi: *Web Service* Maktabah YARSI mengirim respons yang berisi pesan kesalahan dan memberikan informasi tentang kesalahan yang terjadi.

Langkah-langkah:

1. Terjadi kesalahan dalam proses penambahan kategori buku.
2. *Web Service* Maktabah YARSI mengirim respons yang berisi pesan kesalahan kepada pengguna melalui *Postman*.
3. Pengguna melihat pesan kesalahan dan dapat melakukan perbaikan atau tindakan selanjutnya sesuai dengan informasi yang diberikan.

1.4.29 Fitur Pengelola Data Menambah Sub Kategori Buku

1.4.29.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk menambahkan sub kategori buku baru ke dalam sistem. Sub kategori buku ini memerlukan beberapa atribut, termasuk *_id* sebagai *primary key*, *name* sebagai nama sub kategori, *total* sebagai total buku dalam sub kategori, dan kategori sebagai referensi ke kategori buku utama.

1.4.29.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan POST melalui *Postman* atau alat pengelola API lainnya untuk menambahkan sub kategori buku baru.

1.4.29.3 Input

Data yang diperlukan untuk menambah sub kategori buku melibatkan parameter sebagai berikut:

1. `_id` (objectId) [*Primary key*]: ID unik sebagai kunci utama.
2. `name` (string): Nama *sub kategori* buku.
3. `total` (string): Total buku dalam *sub kategori*.
4. `kategori` (objectId): ID referensi ke kategori buku utama.

1.4.29.4 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang memberikan konfirmasi penambahan sub kategori buku atau memberikan pesan kesalahan jika ada.

1.4.29.5 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang diperlukan.

Pascakondisi: Sub kategori buku baru berhasil ditambahkan ke dalam sistem.

Langkah-langkah:

1. Pengguna membuat permintaan POST ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter yang dibutuhkan, seperti `_id` (objectId), `name`, `total`, dan `kategori`.
3. *Web Service* Maktabah YARSI memproses permintaan dan menambahkan *sub kategori* buku baru ke dalam sistem.
4. *Web Service* Maktabah YARSI mengirim respons yang berisi konfirmasi penambahan *sub kategori* buku atau pesan kesalahan jika ada.
5. Pengguna melihat respons dari *Postman* yang menunjukkan status penambahan *sub kategori* buku.

1.4.30 Fitur Pengelola Data Menambah Buku

1.4.30.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk menambahkan buku baru ke dalam sistem. Buku ini

memerlukan beberapa atribut, termasuk *_id* sebagai *primary key*, *title* sebagai judul buku, *description* sebagai deskripsi buku, *contributor* sebagai kontributor buku, *creator* sebagai pembuat buku, *publisher* sebagai penerbit buku, *createdAt* sebagai tanggal buku dibuat, *kategori_id* sebagai referensi ke kategori buku, *sub kategori_id* sebagai referensi ke *sub kategori* buku, *total* sebagai total buku, *page* sebagai jumlah halaman buku, dan *source* sebagai sumber buku.

1.4.30.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan POST melalui *Postman* atau alat pengelola API lainnya untuk menambahkan buku baru.

1.4.30.3 Input

Data yang diperlukan untuk menambah buku melibatkan parameter sebagai berikut:

1. *_id* (objectId) [*Primary key*]: ID unik sebagai kunci utama.
2. *title* (string): Judul buku.
3. *description* (string): Deskripsi buku.
4. *contributor* (string): Kontributor buku.
5. *creator* (string): Pembuat buku.
6. *publisher* (string): Penerbit buku.
7. *createdAt* (Date): Tanggal buku dibuat.
8. *kategori_id* (objectId): ID referensi ke kategori buku.
9. *sub kategori_id* (objectId): ID referensi ke *sub kategori* buku.
10. *total* (number): Total buku.
11. *page* (number): Jumlah halaman buku.
12. *source* (string): Sumber buku.

1.4.30.4 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang memberikan konfirmasi penambahan buku atau memberikan pesan kesalahan jika ada.

1.4.30.5 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang diperlukan.

Pascakondisi: Buku baru berhasil ditambahkan ke dalam sistem.

Langkah-langkah:

1. Pengguna membuat permintaan POST ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter yang dibutuhkan, seperti *_id* (*objectId*), *title*, *description*, *contributor*, *creator*, *publisher*, *createdAt*, *kategori_id*, *sub kategori_id*, *total*, dan *page*.
3. *Web Service* Maktabah YARSI memproses permintaan dan menambahkan buku baru ke dalam sistem.
4. *Web Service* Maktabah YARSI mengirim respons yang berisi konfirmasi penambahan buku atau pesan kesalahan jika ada.
5. Pengguna melihat respons dari *Postman* yang menunjukkan status penambahan buku.

1.4.31 Fitur Pengelola Data Menambah Tabel of Content

1.4.31.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk menambahkan daftar isi (Tabel of Content) untuk sebuah buku. Daftar isi ini memerlukan beberapa atribut, termasuk *_id* sebagai *primary key*, *page* sebagai nomor halaman di dalam buku, *name* sebagai nama entri dalam daftar isi, *Bibliography* sebagai referensi ke buku yang terkait, dan *sub* sebagai array yang berisi sub-entri dalam daftar isi.

1.4.31.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan POST melalui *Postman* atau alat pengelola API lainnya untuk menambahkan daftar isi untuk sebuah buku.

1.4.31.3 Input

Data yang diperlukan untuk menambah daftar isi melibatkan parameter sebagai berikut:

1. *_id* (*objectId*) [*Primary key*]: ID unik sebagai kunci utama.
2. *page* (*number*): Nomor halaman di dalam buku.
3. *name* (*string*): Nama entri dalam daftar isi.
4. *Bibliography* (*objectId*): ID referensi ke buku terkait.
5. *sub* (*array*): Array yang berisi sub-entri dalam daftar isi.

1.4.31.4 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang memberikan konfirmasi penambahan daftar isi atau memberikan pesan kesalahan jika ada.

1.4.31.5 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang diperlukan.

Pascakondisi: Daftar isi baru berhasil ditambahkan untuk sebuah buku.

Langkah-langkah:

1. Pengguna membuat permintaan POST ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter yang dibutuhkan, seperti *_id* (*objectId*), *page*, *name*, *bibliography*, dan *sub*.
3. *Web Service* Maktabah YARSI memproses permintaan dan menambahkan daftar isi baru untuk sebuah buku.
4. *Web Service* Maktabah YARSI mengirim respons yang berisi konfirmasi penambahan daftar isi atau pesan kesalahan jika ada.
5. Pengguna melihat respons dari *Postman* yang menunjukkan status penambahan daftar isi.

1.4.32 Fitur Pengelola Data Menambah Konten Buku

1.4.32.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk menambahkan konten buku. Konten buku ini memerlukan beberapa atribut, termasuk *_id* sebagai *primary key*, *heading* sebagai judul bagian konten, *listcontent* sebagai referensi ke daftar isi buku, *page* sebagai nomor halaman di dalam buku, dan *text* sebagai teks konten buku.

1.4.32.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan POST melalui *Postman* atau alat pengelola API lainnya untuk menambahkan konten untuk sebuah buku.

1.4.32.3 *Input*

Data yang diperlukan untuk menambah konten buku melibatkan parameter sebagai berikut:

1. `_id` (objectId) [*Primary key*]: ID unik sebagai kunci utama.
2. `heading` (string): Judul bagian konten buku.
3. `listcontent` (objectId): ID referensi ke daftar isi buku terkait.
4. `page` (number): Nomor halaman di dalam buku.
5. `text` (string): Teks konten buku.

1.4.32.4 *Output*

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang memberikan konfirmasi penambahan konten buku atau memberikan pesan kesalahan jika ada.

1.4.32.5 *Skenario Utama*

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang dibutuhkan.

Pascakondisi: Konten buku baru berhasil ditambahkan untuk sebuah buku.

Langkah-langkah:

1. Pengguna membuat permintaan POST ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter yang dibutuhkan, seperti `_id` (objectId), `heading`, `listcontent`, `page`, dan `text`.
3. *Web Service* Maktabah YARSI memproses permintaan dan menambahkan konten buku baru untuk sebuah buku.
4. *Web Service* Maktabah YARSI mengirim respons yang berisi konfirmasi penambahan konten buku atau pesan kesalahan jika ada.
5. Pengguna melihat respons dari *Postman* yang menunjukkan status penambahan konten buku.

1.4.33 *Fitur Pengelola Data Menghapus Kategori Buku*

1.4.33.1 *Deskripsi*

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk menghapus kategori buku dari sistem.

1.4.33.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan DELETE melalui *Postman* atau alat pengelola API lainnya untuk menghapus kategori buku tertentu.

1.4.33.3 Input

Data yang diperlukan untuk menghapus kategori buku melibatkan parameter `_id` (objectId) [*Primary key*]: ID unik dari kategori buku yang akan dihapus.

1.4.33.4 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang memberikan konfirmasi penghapusan kategori buku atau memberikan pesan kesalahan jika ada.

1.4.33.5 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki akses token atau otorisasi yang valid.

Pascakondisi: Kategori buku tertentu berhasil dihapus dari sistem.

Langkah-langkah:

1. Pengguna membuat permintaan DELETE ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter yang dibutuhkan, seperti `_id` (objectId) dari kategori buku yang akan dihapus.
3. *Web Service* Maktabah YARSI memproses permintaan dan menghapus kategori buku yang sesuai dengan ID yang diberikan.
4. *Web Service* Maktabah YARSI mengirim respons yang berisi konfirmasi penghapusan kategori buku atau pesan kesalahan jika ada.
5. Pengguna melihat respons dari *Postman* yang menunjukkan status penghapusan kategori buku.

1.4.34 Fitur Pengelola Data Menghapus Sub Kategori Buku

1.4.34.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk menghapus sub kategori buku dari sistem.

1.4.34.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan DELETE melalui *Postman* atau alat pengelola API lainnya untuk menghapus sub kategori buku tertentu.

1.4.34.3 Input

Data yang diperlukan untuk menghapus sub kategori buku melibatkan parameter `_id` (objectId) [*Primary key*]: ID unik dari sub kategori buku yang akan dihapus.

1.4.34.4 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang memberikan konfirmasi penghapusan sub kategori buku atau memberikan pesan kesalahan jika ada.

1.4.34.5 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang dibutuhkan.

Pascakondisi: Sub kategori buku tertentu berhasil dihapus dari sistem.

Langkah-langkah:

1. Pengguna membuat permintaan DELETE ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter yang dibutuhkan, seperti `_id` (objectId) dari sub kategori buku yang akan dihapus.
3. *Web Service* Maktabah YARSI memproses permintaan dan menghapus sub kategori buku yang sesuai dengan ID yang diberikan.
4. *Web Service* Maktabah YARSI mengirim respons yang berisi konfirmasi penghapusan sub kategori buku atau pesan kesalahan jika ada.
5. Pengguna melihat respons dari *Postman* yang menunjukkan status penghapusan sub kategori buku.

1.4.35 Fitur Pengelola Data Menghapus Buku

1.4.35.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk menghapus buku dari sistem.

1.4.35.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan DELETE melalui *Postman* atau alat pengelola API lainnya untuk menghapus buku tertentu.

1.4.35.3 Input

Data yang diperlukan untuk menghapus buku melibatkan parameter `_id` (`objectId`) [*Primary key*]: ID unik dari buku yang akan dihapus.

1.4.35.4 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang memberikan konfirmasi penghapusan buku atau memberikan pesan kesalahan jika ada.

1.4.35.5 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang dibutuhkan.

Pascakondisi: Buku tertentu berhasil dihapus dari sistem.

Langkah-langkah:

1. Pengguna membuat permintaan DELETE ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter yang dibutuhkan, seperti `_id` (`objectId`) dari buku yang akan dihapus.
3. *Web Service* Maktabah YARSI memproses permintaan dan menghapus buku yang sesuai dengan ID yang diberikan.
4. *Web Service* Maktabah YARSI mengirim respons yang berisi konfirmasi penghapusan buku atau pesan kesalahan jika ada.
5. Pengguna melihat respons dari *Postman* yang menunjukkan status penghapusan buku.

1.4.36 Fitur Pengelola Data Menghapus Tabel of Content

1.4.36.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk menghapus daftar isi (Tabel of Content) dari sebuah buku.

1.4.36.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan DELETE melalui *Postman* atau alat pengelola API lainnya untuk menghapus daftar isi dari sebuah buku.

1.4.36.3 Input

Data yang diperlukan untuk menghapus daftar isi melibatkan parameter `_id` (objectId) [*Primary key*]: ID unik dari daftar isi yang akan dihapus.

1.4.36.4 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang memberikan konfirmasi penghapusan daftar isi atau memberikan pesan kesalahan jika ada.

1.4.36.5 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang dibutuhkan.

Pascakondisi: Daftar isi dari sebuah buku berhasil dihapus dari sistem.

Langkah-langkah:

1. Pengguna membuat permintaan DELETE ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter yang dibutuhkan, seperti `_id` (objectId) dari daftar isi yang akan dihapus.
3. *Web Service* Maktabah YARSI memproses permintaan dan menghapus daftar isi yang sesuai dengan ID yang diberikan.
4. *Web Service* Maktabah YARSI mengirim respons yang berisi konfirmasi penghapusan daftar isi atau pesan kesalahan jika ada.
5. Pengguna melihat respons dari *Postman* yang menunjukkan status penghapusan daftar isi.

1.4.37 Fitur Pengelola Data: Menghapus Konten Buku

1.4.37.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk menghapus konten buku dari sebuah buku.

1.4.37.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan DELETE melalui *Postman* atau alat pengelola API lainnya untuk menghapus konten buku dari sebuah buku.

1.4.37.3 Input

Data yang diperlukan untuk menghapus konten buku melibatkan parameter `_id` (`objectId`) [*Primary key*]: ID unik dari konten buku yang akan dihapus.

1.4.37.4 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang memberikan konfirmasi penghapusan konten buku atau memberikan pesan kesalahan jika ada.

1.4.37.5 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang diperlukan

Pascakondisi: Konten buku dari sebuah buku berhasil dihapus dari sistem.

Langkah-langkah:

1. Pengguna membuat permintaan DELETE ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter yang dibutuhkan, seperti `_id` (`objectId`) dari konten buku yang akan dihapus.
3. *Web Service* Maktabah YARSI memproses permintaan dan menghapus konten buku yang sesuai dengan ID yang diberikan.
4. *Web Service* Maktabah YARSI mengirim respons yang berisi konfirmasi penghapusan konten buku atau pesan kesalahan jika ada.
5. Pengguna melihat respons dari *Postman* yang menunjukkan status penghapusan konten buku.

1.4.38 Fitur Pengelola Data Melihat Kategori Buku

1.4.38.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk melihat daftar kategori buku yang terdapat dalam sistem.

1.4.38.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan GET melalui *Postman* atau alat pengelola API lainnya untuk melihat daftar kategori buku.

1.4.38.3 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang berisi daftar kategori buku atau pesan kesalahan jika ada.

1.4.38.4 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang dibutuhkan.

Pascakondisi: Pengguna melihat daftar kategori buku yang terdapat dalam sistem.

Langkah-langkah:

1. Pengguna membuat permintaan GET ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. *Web Service* Maktabah YARSI memproses permintaan dan mengembalikan respons yang berisi daftar kategori buku.
3. Pengguna melihat daftar kategori buku pada respons *Postman*.

1.4.39 Fitur Pengelola Data Melihat Sub Kategori Buku

1.4.39.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk melihat daftar sub kategori buku yang terdapat dalam sistem.

1.4.39.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan GET melalui *Postman* atau alat pengelola API lainnya untuk melihat daftar sub kategori buku.

1.4.39.3 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang berisi daftar sub kategori buku atau pesan kesalahan jika ada.

1.4.39.4 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang dibutuhkan.

Pascakondisi: Pengguna melihat daftar sub kategori buku yang terdapat dalam sistem.

Langkah-langkah:

1. Pengguna membuat permintaan GET ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. *Web Service* Maktabah YARSI memproses permintaan dan mengembalikan respons yang berisi daftar sub kategori buku.
3. Pengguna melihat daftar sub kategori buku pada respons *Postman*.

1.4.40 Fitur Pengelola Data Melihat Buku

1.4.40.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk melihat daftar buku berdasarkan nama kategori tertentu dan ID buku yang terdapat dalam sistem.

1.4.40.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan GET melalui *Postman* atau alat pengelola API lainnya untuk melihat daftar buku berdasarkan nama kategori dan ID buku.

1.4.40.3 Input

Data yang diperlukan untuk fitur ini melibatkan parameter sebagai berikut:

1. *kategori_name* (string): Nama kategori buku yang ingin dilihat.
2. *bibliography_id* (objectId) [*Primary key*]: ID unik dari buku yang ingin dilihat.

1.4.40.4 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang berisi daftar buku berdasarkan nama kategori dan ID buku atau pesan kesalahan jika ada.

1.4.40.5 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang dibutuhkan.

Pascakondisi: Pengguna melihat daftar buku berdasarkan nama kategori dan ID buku yang terdapat dalam sistem.

Langkah-langkah:

1. Pengguna membuat permintaan GET ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter *kategori_name* yang merupakan nama kategori buku yang ingin dilihat.
3. Pengguna menyertakan parameter *book_id* yang merupakan ID unik dari buku yang ingin dilihat.
4. *Web Service* Maktabah YARSI memproses permintaan dan mengembalikan respons yang berisi daftar buku berdasarkan nama kategori dan ID buku.
5. Pengguna melihat daftar buku pada respons *Postman*.

1.4.41 Fitur Pengelola Data Melihat Tabel of Content

1.4.41.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk melihat daftar Tabel of Content (TOC) berdasarkan ID buku yang terdapat dalam sistem.

1.4.41.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan GET melalui *Postman* atau alat pengelola API lainnya untuk melihat daftar TOC berdasarkan ID buku.

1.4.41.3 Input

Data yang diperlukan untuk fitur ini melibatkan parameter *Bibliography._id* (*objectId*) [*Primary key*]: ID unik dari buku yang ingin dilihat TOC-nya.

1.4.41.4 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang berisi daftar TOC berdasarkan ID buku atau pesan kesalahan jika ada.

1.4.41.5 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang diperlukan.

Pascakondisi: Pengguna melihat daftar TOC berdasarkan ID buku yang terdapat dalam sistem.

Langkah-langkah:

1. Pengguna membuat permintaan GET ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter *book_id* yang merupakan ID unik dari buku yang ingin dilihat TOC-nya.
3. *Web Service* Maktabah YARSI memproses permintaan dan mengembalikan respons yang berisi daftar TOC berdasarkan ID buku.
4. Pengguna melihat daftar TOC pada respons *Postman*.

1.4.42 Fitur Pengelola Data Melihat Konten Buku

1.4.42.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk melihat daftar konten buku berdasarkan *list content* yang terdapat dalam sistem.

1.4.42.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan GET melalui *Postman* atau alat pengelola API lainnya untuk melihat daftar konten buku berdasarkan *list content*.

1.4.42.3 Input

Data yang diperlukan untuk fitur ini melibatkan parameter *TabelContent._id* (*objectId*) yaitu ID unik dari *list content* yang ingin dilihat kontennya.

1.4.42.4 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang berisi daftar konten buku berdasarkan *list content* atau pesan kesalahan jika ada.

1.4.42.5 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang diperlukan.

Pascakondisi: Pengguna melihat daftar konten buku berdasarkan *list* content yang terdapat dalam sistem.

Langkah-langkah:

1. Pengguna membuat permintaan GET ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter *list_content* yang merupakan ID unik dari *list* content yang ingin dilihat kontennya.
3. *Web Service* Maktabah YARSI memproses permintaan dan mengembalikan respons yang berisi daftar konten buku berdasarkan *list* content.
4. Pengguna melihat daftar konten buku pada respons *Postman*.

1.4.43 Fitur Pengelola Data Menyunting Kategori Buku

1.4.43.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk menyunting informasi kategori buku yang telah ada dalam sistem.

1.4.43.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan PATCH melalui *Postman* atau alat pengelola API lainnya untuk menyunting kategori buku.

1.4.43.3 Input

Data yang diperlukan untuk fitur ini melibatkan parameter *kategori_id* (*objectId*) [*Primary key*] yaitu ID unik dari kategori buku yang ingin disunting.

Data yang ingin disunting, seperti:

1. *name* (string): Nama baru untuk kategori buku.
2. *total* (number): Total buku baru dalam kategori.

1.4.43.4 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang berisi informasi terkini tentang kategori buku yang telah disunting atau pesan kesalahan jika ada.

1.4.43.5 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang diperlukan.

Pascakondisi: Informasi kategori buku dalam sistem telah diperbarui sesuai dengan permintaan pengguna.

Langkah-langkah:

1. Pengguna membuat permintaan PATCH ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter *kategori_id* yang merupakan ID unik dari kategori buku yang ingin disunting.
3. Pengguna menyertakan data yang ingin disunting, seperti *name* (nama baru) dan *total* (total buku baru).
4. *Web Service* Maktabah YARSI memproses permintaan dan mengembalikan respons yang berisi informasi terkini tentang kategori buku yang telah disunting.
5. Pengguna melihat informasi terkini pada respons *Postman*.

1.4.44 Fitur Pengelola Data Menyunting Sub Kategori Buku

1.4.44.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk menyunting informasi sub kategori buku yang telah ada dalam sistem.

1.4.44.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan PATCH melalui *Postman* atau alat pengelola API lainnya untuk menyunting sub kategori buku.

1.4.44.3 Input

Data yang diperlukan untuk fitur ini melibatkan parameter sebagai berikut:

sub kategori_id (objectId) [*Primary key*]: ID unik dari sub kategori buku yang ingin disunting.

Data yang ingin disunting, seperti:

1. *name* (string): Nama baru untuk sub kategori buku.
2. *total* (number): Total buku baru dalam sub kategori.

1.4.44.4 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang berisi informasi terkini tentang sub kategori buku yang telah disunting atau pesan kesalahan jika ada.

1.4.44.5 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki akses token atau otorisasi yang valid.

Pascakondisi: Informasi sub kategori buku dalam sistem telah diperbarui sesuai dengan permintaan pengguna.

Langkah-langkah:

1. Pengguna membuat permintaan *PATCH* ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter *sub kategori_id* yang merupakan ID unik dari sub kategori buku yang ingin disunting.
3. Pengguna menyertakan data yang ingin disunting, seperti *name* (nama baru) dan *total* (total buku baru).
4. *Web Service* Maktabah YARSI memproses permintaan dan mengembalikan respons yang berisi informasi terkini tentang sub kategori buku yang telah disunting.
5. Pengguna melihat informasi terkini pada respons *Postman*.

1.4.45 Fitur Pengelola Data Menyunting Buku

1.4.45.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk menyunting informasi buku yang telah ada dalam sistem.

1.4.45.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan PATCH melalui *Postman* atau alat pengelola API lainnya untuk menyunting buku.

1.4.45.3 Input

Data yang diperlukan untuk fitur ini melibatkan parameter `book_id` (objectId) [*Primary key*] yaitu ID unik dari buku yang ingin disunting.

Data yang ingin disunting, seperti:

1. `title` (string): Judul baru untuk buku.
2. `description` (string): Deskripsi baru untuk buku.
3. `contributor` (string): Kontributor baru untuk buku.
4. `creator` (string): Pencipta baru untuk buku.
5. `publisher` (string): Penerbit baru untuk buku.
6. `kategori_id` (objectId): ID kategori baru untuk buku.
7. `sub kategori_id` (objectId): ID sub kategori baru untuk buku.
8. `total` (number): Total baru untuk buku.
9. `page` (number): Jumlah halaman baru untuk buku.

1.4.45.4 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang berisi informasi terkini tentang buku yang telah disunting atau pesan kesalahan jika ada.

1.4.45.5 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang dibutuhkan.

Pascakondisi: Informasi buku dalam sistem telah diperbarui sesuai dengan permintaan pengguna.

Langkah-langkah:

1. Pengguna membuat permintaan PATCH ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter `book_id` yang merupakan ID unik dari buku yang ingin disunting.

3. Pengguna menyertakan data yang ingin disunting, seperti title, description, contributor, creator, publisher, kategori_id, sub kategori_id, total, dan page.
4. *Web Service* Maktabah YARSI memproses permintaan dan mengembalikan respons yang berisi informasi terkini tentang buku yang telah disunting.
5. Pengguna melihat informasi terkini pada respons *Postman*.

1.4.46 Fitur Pengelola Data Menyunting *Table of Content*

1.4.46.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk menyunting informasi tabel isi (*Table of Content*) buku yang telah ada dalam sistem.

1.4.46.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan PATCH melalui *Postman* atau alat pengelola API lainnya untuk menyunting tabel isi buku.

1.4.46.3 Input

Data yang diperlukan untuk fitur ini sebagai berikut:

1. page (number): Nomor halaman baru untuk tabel isi buku.
2. name (string): Nama baru untuk entri tabel isi buku.
3. sub (array): Array baru untuk sub entri tabel isi buku.

1.4.46.4 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang berisi informasi terkini tentang tabel isi buku yang telah disunting atau pesan kesalahan jika ada.

1.4.46.5 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang diperlukan.

Pascakondisi: Informasi tabel isi buku dalam sistem telah diperbarui sesuai dengan permintaan pengguna.

Langkah-langkah:

1. Pengguna membuat permintaan PATCH ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter *toc_id* yang merupakan ID unik dari tabel isi buku yang ingin disunting.
3. Pengguna menyertakan data yang ingin disunting, seperti *page*, *name*, dan *sub*.
4. *Web Service* Maktabah YARSI memproses permintaan dan mengembalikan respons yang berisi informasi terkini tentang tabel isi buku yang telah disunting.
5. Pengguna melihat informasi terkini pada respons *Postman*.

1.4.47 Fitur Pengelola Data Menyunting Konten Buku

1.4.47.1 Deskripsi

Fitur Pengelola Data pada *Web Service* Maktabah YARSI memungkinkan pengguna dengan peran pengelola untuk menyunting konten buku yang telah ada dalam sistem.

1.4.47.2 Trigger

Fitur ini diaktifkan ketika pengguna dengan peran pengelola membuat permintaan PATCH melalui *Postman* atau alat pengelola API lainnya untuk menyunting konten buku.

1.4.47.3 Input

Data yang diperlukan untuk fitur ini sebagai berikut:

1. *heading* (string): Judul baru untuk konten buku.
2. *listcontent* (TabelContent) [ref: - TabelContent._id]: ID baru untuk isi konten buku.
3. *page* (number): Nomor halaman baru untuk konten buku.
4. *text* (string): Teks baru untuk konten buku.

1.4.47.4 Output

Keluaran dari fitur ini adalah respons dari *Web Service* Maktabah YARSI yang berisi informasi terkini tentang konten buku yang telah disunting atau pesan kesalahan jika ada.

1.4.47.5 Skenario Utama

Prakondisi: Pengguna dengan peran pengelola membuka *Postman* dan memiliki *endpoint* yang diperlukan.

Pascakondisi: Informasi konten buku dalam sistem telah diperbarui sesuai dengan permintaan pengguna.

Langkah-langkah:

1. Pengguna membuat permintaan PATCH ke *endpoint* yang disediakan oleh *Web Service* Maktabah YARSI melalui *Postman*.
2. Pengguna menyertakan parameter *content_id* yang merupakan ID unik dari konten buku yang ingin disunting.
3. Pengguna menyertakan data yang ingin disunting, seperti *heading*, *listcontent*, *page*, dan *text*.
4. *Web Service* Maktabah YARSI memproses permintaan dan mengembalikan respons yang berisi informasi terkini tentang konten buku yang telah disunting.
5. Pengguna melihat informasi terkini pada respons *Postman*.

1.5 Kebutuhan Nonfungsional

1.5.1 Atribut Kualitas

Aplikasi android *mobile* Maktabah YARSI memiliki atribut kualitas sebagai berikut:

1. Keamanan Sistem (*System Security*)

Deskripsi: Atribut ini mencakup upaya untuk melindungi Maktabah YARSI dari akses tidak sah dan menjaga integritas data.

Batas Minimum: Sistem harus mematuhi standar keamanan ISO/IEC 27001:2013 dan melibatkan enkripsi data sensitif.

Pengaruh pada Bagian Sistem: Keamanan sistem mempengaruhi seluruh sistem dengan menetapkan kebijakan keamanan, terutama pada manajemen akses dan enkripsi data.

2. Kemudahan Pemeliharaan (*Maintainability*)

Deskripsi: Atribut ini menitikberatkan pada kemudahan pemeliharaan dan pengembangan sistem.

Batas Minimum: Kode sumber sistem harus terdokumentasi dengan baik dan menggunakan praktik pengembangan yang bersih.

Pengaruh pada Bagian Sistem: Kemudahan pemeliharaan memengaruhi seluruh sistem, terutama pada pengembangan modul baru dan perbaikan *bug*.

1.5.2 Kebutuhan Legal

Pengembangan Maktabah YARSI sudah disepakati melalui Surat Perjanjian Implementasi antara Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas YARSI dengan Klien.

dengan:

No. Program Studi Teknik Informatika: 024/Prodi TI/HK.20/X/2023 yang berisikan aturan yang harus dipenuhi dalam aplikasi. Data yang didapat untuk diolah dan ditampilkan dalam Maktabah YARSI menggunakan buku yang sudah diwaqafkan dari Rumah Fiqih.