

Rancang Bangun Aplikasi MakotaMu Berbasis Android (Kasus: Sistem Informasi Majelis Tabligh PDM Kota Malang)

Muhammad Syofi Azmi^{*1}, Ilyas Nuryasin², Evi Dwi Wahyuni³

^{1,2,3}Universitas Muhammadiyah Malang

syofiazmi@webmail.umm.ac.id^{*1}, ilyas@umm.ac.id², evi.dwi@umm.ac.id³

Abstrak

Ketersediaan informasi jadwal khatib masjid Muhammadiyah Malang dan booklet Uswatun Hasanah merupakan hal penting untuk memperluas dakwah Muhammadiyah yang ada di Malang. Oleh karena itu, tugas akhir ini ada untuk berusaha menjadi solusi dengan memanfaatkan teknologi yang ada untuk membuat platform berupa aplikasi berbasis android yang menjadi jembatan informasi seputar Muhammadiyah Malang. Penulis menggunakan metode incremental dalam membangun aplikasi di mana fungsi prioritas akan dibuat terlebih dahulu sebelum fitur-fitur pelengkap lain dibuat. Kemudian untuk mendesain aplikasi MakotaMu penulis menggunakan metode User Centered Design (UCD) dengan alat dokumen requirement checklist, di mana dokumen ini akan diperiksa dan divalidasi langsung oleh stakeholder. Aplikasi yang diberi nama MakotaMu ini diuji menggunakan metode black box testing yang menghasilkan aplikasi dapat berfungsi dengan baik dan dapat digunakan oleh pengguna. Berdasarkan hasil pengujian usability menggunakan kuesioner SUS di mana aplikasi MakotaMu mendapatkan skor 78 dengan grade "B" termasuk dalam kategori aplikasi yang acceptable, maka dapat disimpulkan bahwa aplikasi MakotaMu dapat diterima dan mudah digunakan.

Kata Kunci: informasi, Incremental, User Centered Design, Black-Box, System Usability Scale

Abstract

The availability of information on the schedule of preachers of the Muhammadiyah Malang mosque and the Uswatun Hasanah booklet is important to expand the preaching of Muhammadiyah in Malang. Therefore, this final project is to try to be a solution by utilizing existing technology to create a platform in the form of an android-based application that becomes a bridge of information about Muhammadiyah Malang. The author uses the incremental method in building applications where the priority function will be created first before other complementary features are created. Then to design the MakotaMu application, the author uses the User Centered Design (UCD) method with the requirements checklist document tool, where this document will be checked and validated directly by stakeholders. The application, named MakotaMu, is tested using the black box testing method which results in the application being able to function properly and can be used by users. Based on the results of usability testing using the SUS questionnaire where the MakotaMu application gets a score of 78 with a grade of "B" which is included in the acceptable application category, it can be concluded that the MakotaMu application is acceptable and easy to use.

Keywords: Information, Incremental, User Centered Design, Black-Box, System Usability Scale

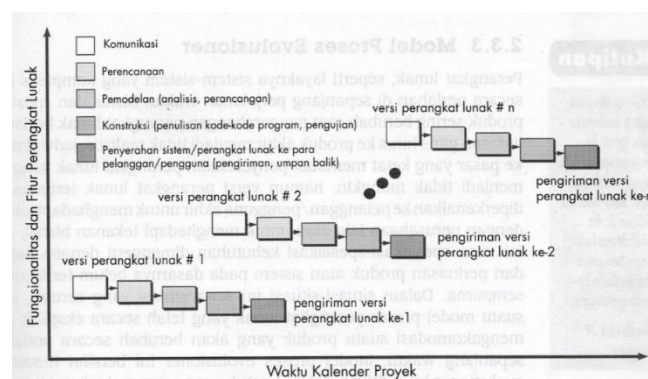
1. Pendahuluan

Majelis Tabligh adalah salah satu majelis yang terdapat dalam struktur Pimpinan Daerah Muhammadiyah (PDM) kota Malang. Majelis Tabligh ini mempunyai bagian yaitu Corps Mubaligh Muhammadiyah (CMM) Malang. CMM ini memiliki fungsi dan tugas antara lain, memberikan layanan masyarakat akan kebutuhan tenaga Mubaligh/Mubalighat, menggerakkan dan mengelola kegiatan dakwah dan tabligh Muhammadiyah, meningkatkan dan mengembangkan kompetensi Mubaligh Muhammadiyah, mengelola pola kesejahteraan Mubaligh Muhammadiyah, berperan aktif dalam memakmurkan masjid-masjid Muhammadiyah dan masjid lain yang bersedia atau meminta bergabung dalam koordinasi Muhammadiyah, bersama Majelis Tabligh menyelenggarakan pembinaan, pendidikan dan pelatihan Mubaligh/Mubalighat Muhammadiyah, melakukan koordinasi tugas dan kegiatan Mubaligh Muhammadiyah sesuai tingkat dan proporsinya masing-masing, menyiarkan pokok-pokok pikiran, ideologi dan paham agama,

Muhammadiyah[1]. Sebagaimana fungsi CMM yang telah disebutkan di atas, CMM telah melakukan beberapa bentuk kegiatan yaitu dengan menerbitkan buklet yang bernama Uswatun Hasanah. Buklet ini dibagikan setiap hari jum'at di mesjid-mesjid Muhammadiyah yang ada di Malang. Kemudian bentuk kegiatan lainnya yaitu melakukan musyawarah untuk merumuskan jadwal khatib shalat jum'at setiap 4 (empat) bulan sekali. Musyawarah ini melibatkan seluruh mubaligh dan takmir masjid Muhammadiyah yang ada di Malang. Untuk memperluas dakwah Muhammadiyah di Malang, maka diusulkan untuk membuat sebuah aplikasi 'MakotaMu' yang dapat menyediakan informasi seputar Muhammadiyah yang ada di Malang. Sehingga buklet uswatun hasanah yang selama ini hanya berbentuk buku kecil yang hanya menjangkau setiap masjid Muhammadiyah yang ada di Malang, dapat dimasukkan ke dalam sebuah platform berupa aplikasi 'MakotaMu' berbasis android yang diharapkan dapat menjangkau setiap masyarakat pada umumnya dan warga Muhammadiyah Malang pada khususnya. Selain buklet uswatun hasanah, pada aplikasi tersebut juga dapat diisi dengan informasi bermanfaat lainnya seperti, jadwal khatib, daftar mubaligh Muhammadiyah (CMM), daftar masjid Muhammadiyah (MasjidMu), sekolah Muhammadiyah Malang (Dikdasmen) serta informasi lainnya yang diinginkan. Karena sebelumnya belum ada aplikasi yang menyediakan informasi seputar Muhammadiyah Malang, maka dibuatnya aplikasi 'MakotaMu' ini diharapkan dapat membantu Majelis Tabligh dan CMM dalam menyediakan informasi seputar Muhammadiyah Malang baik untuk organisasi maupun masyarakat Malang pada umumnya.

2. Metode Penelitian

Metode pengembangan perangkat lunak yang digunakan adalah metode *incremental*. Metode *incremental* menurut Roger S. Pressman dalam bukunya menjelaskan bahwa model pengembangan perangkat lunak jenis ini bisa digunakan ketika situasi di mana spesifikasi kebutuhan perangkat lunak sudah terdefinisi dengan baik di awal, tetapi lingkup keseluruhan usaha pengembangan perangkat lunak tidak bisa dilakukan sekaligus, perangkat lunak akan dibangun dengan kebutuhan yang paling penting terlebih dahulu dan sudah bisa digunakan oleh pengguna. Sehingga perangkat lunak sudah bisa digunakan oleh pengguna secara cepat. Kemudian perangkat lunak akan dikembangkan lagi sesuai kesepakatan pengembang dan pengguna. Jadi pada intinya adalah pengembangan perangkat lunak dengan menggunakan model seperti ini akan menghasilkan perangkat lunak yang cepat dan dapat dilakukan penambahan fungsi sedikit demi sedikit (*incremental*)[2]. Metode ini memiliki 5 (lima) aktivitas antara lain komunikasi, perencanaan, pemodelan, konstruksi dan penyerahan perangkat lunak ke pengguna (*Deployment*). Metode *incremental* dapat dilihat pada Gambar 1 di bawah ini.



Gambar 1. Model Incremental[2]

2.1 Komunikasi

Pada aktivitas ini, penulis melakukan wawancara dengan *stakeholder*. Hasil dari wawancara ini adalah latar belakang dibuatnya aplikasi berbasis android MakotaMu dan spesifikasi yang dibutuhkan pengguna terhadap aplikasi. Setelah mendapat informasi tentang spesifikasi kebutuhan yang telah disampaikan oleh pengguna, selanjutnya penulis membuat dokumen elisitasi agar spesifikasi kebutuhan dapat terdokumentasi dengan baik. Elisitasi yang dilakukan terdiri dari empat tahap, yakni tahap ke-1 untuk mendokumentasikan semua spesifikasi yang telah didapatkan melalui wawancara. Tahap ke-2 untuk mendokumentasikan tingkat kesulitan pembuatan dari fitur yang dibutuhkan serta menentukan kebutuhan yang menjadi

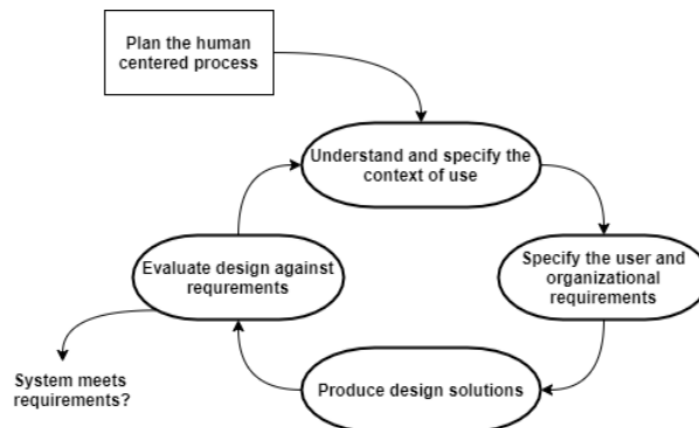
prioritas. Tahap ke-3 untuk mendokumentasikan tingkatan dari tiga aspek yaitu aspek teknik, operasional dan ekonomi. Kemudian tahap final adalah hasil spesifikasi kebutuhan yang telah dipilah dan siap untuk dibuat.

2.2 Perencanaan

Pada aktivitas ini penulis melakukan komunikasi dengan pengguna tentang rencana waktu penyelesaian aplikasi, hasil dari komunikasi tersebut adalah bahwa *stakeholder* tidak menuntut target waktu untuk penyelesaian, dan juga tidak menuntut kesempurnaan aplikasi. Akan tetapi aplikasi wajib untuk dikerjakan menurut kemampuan penulis.

2.3 Pemodelan

Untuk melakukan aktivitas pemodelan ini penulis menggunakan metode *User Centered Design* (UCD). Hal ini dilakukan agar tampilan antarmuka aplikasi yang dibuat sesuai dengan keinginan pengguna, sebagaimana prinsip dari UCD yaitu membuat desain antarmuka yang berfokus pada pengalaman pengguna, seperti pada Gambar 2.



Gambar 2. User Centered Design [3]

Untuk melakukan metode UCD ini terdapat beberapa aktivitas yang perlu dilakukan sebagai berikut [3].

1. *Understand and specify the context of use*, aktivitas ini dilakukan untuk memahami konteks kegunaan aplikasi yang akan dibuat. Seperti pertanyaan siapa yang menggunakan aplikasi nanti dan untuk apa aplikasi digunakan. Konteks kegunaan aplikasi ini bisa didapatkan melalui wawancara dengan *stakeholder* terkait.
2. *Specif the user and organizational requirements*, aktivitas ini dilakukan untuk menentukan spesifikasi apa saja yang dibutuhkan dalam aplikasi. Spesifikasi kebutuhan ini bisa didapatkan melalui komunikasi atau wawancara kepada *stakeholder* kemudian hasil tersebut dimasukkan ke dalam sebuah dokumen yang bernama dokumen elisitasi.
3. *Produce design solutions*, pada aktivitas ini dilakukan perancangan antarmuka. Perancangan ini bisa dimulai dari rancangan kasar seperti coretan di kertas hingga desain yang hampir menyerupai tampilan pada android.
4. *Evaluate design against requirement*, aktivitas ini dilakukan untuk mendapatkan verifikasi dan validasi oleh *stakeholder* yang menyatakan bahwa desain antarmuka yang telah dibuat telah disetujui oleh *stakeholder*. Evaluasi ini bisa dilakukan dengan bantuan dokumen *requirement checklist*.

Ketika desain antarmuka telah disetujui oleh *stakeholder*, maka selanjutnya adalah menerapkan desain tersebut ke dalam penulisan kode program. Di bawah ini adalah hasil dari penerapan desain menggunakan metode UCD.

2.3.1 Analisis Sistem

Setelah mendapatkan spesifikasi kebutuhan perangkat lunak, selanjutnya adalah mengimplementasikan spesifikasi tersebut ke dalam desain perangkat lunak. Gambar 3 di bawah ini adalah gambar spesifikasi aplikasi dalam bentuk diagram *usecase*.

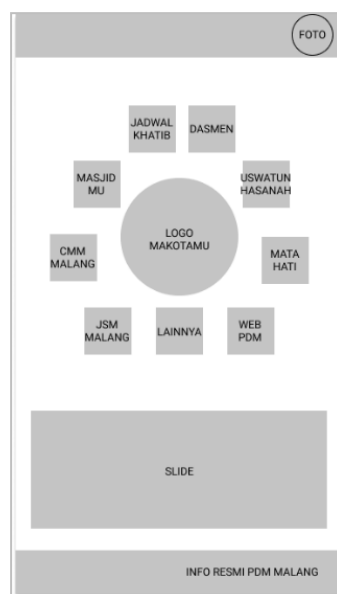


Gambar 3 Usecase Diagram Pengguna

Pengguna memiliki akses untuk *login* atau mendaftar aplikasi, melihat halaman beranda, jadwal khatib, dasmen, uswatun hasanah, web mata hati, web PDM Malang, menu lainnya, JSM, CMM Malang, masjidmu dan info PDM Malang.

2.3.2 Desain Antarmuka

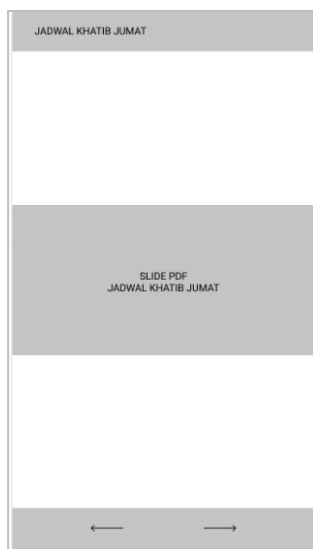
Perancangan antarmuka dilakukan untuk menggambarkan tampilan yang akan dilihat oleh pengguna pada aplikasi. Penulis menggunakan dua cara untuk membuat rancangan antarmuka aplikasi. Pertama adalah perancangan sederhana di mana dalam desainnya hanya ada tulisan, kotak serta warna hitam dan putih. Kedua adalah desain tampilan yang hampir menyerupai tampilan hasil akhir, di mana pada rancangan terdapat gambar, ikon, tulisan dan warna yang beragam. Pada bagian ini penulis hanya akan menampilkan desain sederhana yang memberikan informasi tata letak menu dan tulisan pada sistem. Sedangkan desain yang hampir menyerupai tampilan asli pada android dimasukkan ke dalam dokumen requirement checklist yang akan diverifikasi dan divalidasi oleh *stakeholder*. Desain antarmuka aplikasi terdiri dari beberapa halaman fitur utama antara lain desain antarmuka halaman beranda, jadwal khatib, dasmen, uswatun hasanah, JSM, CMM, masjidmu dan info resmi PDM Malang.



Gambar 4 Desain Halaman Antarmuka Beranda

Keterangan:

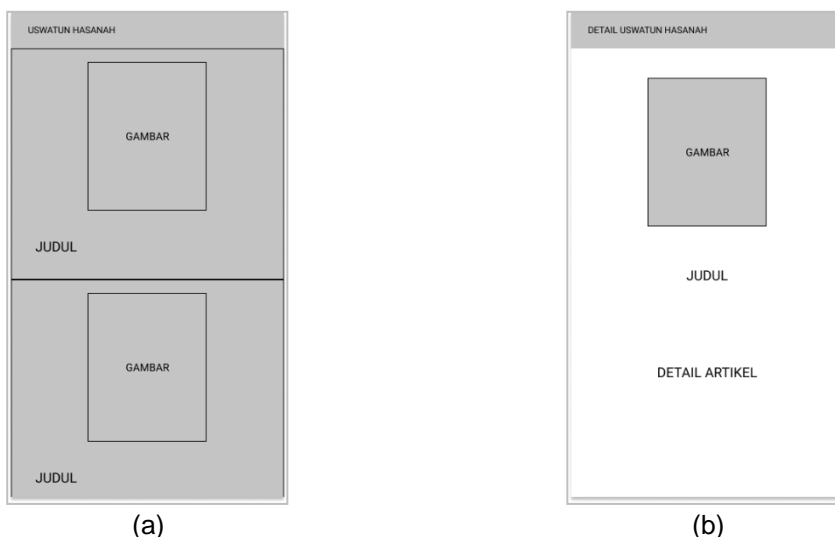
Gambar 4 di atas adalah desain halaman beranda yang akan ditampilkan kepada pengguna. Di sana terdapat menu profil pada sisi kanan atas halaman. Kemudian di sekeliling logo MakotaMu terdapat 9 (sembilan) menu pilihan. Di bawahnya terdapat *slide* yang bisa digunakan untuk menampilkan gambar. Di bawah slide terdapat tulisan yang dapat diinteraksi yaitu Info Resmi PDM Malang yang akan membawa pengguna ke halaman Info Resmi PDM Malang.



Gambar 5. Desain Halaman Antarmuka Jadwal Khatib

Keterangan:

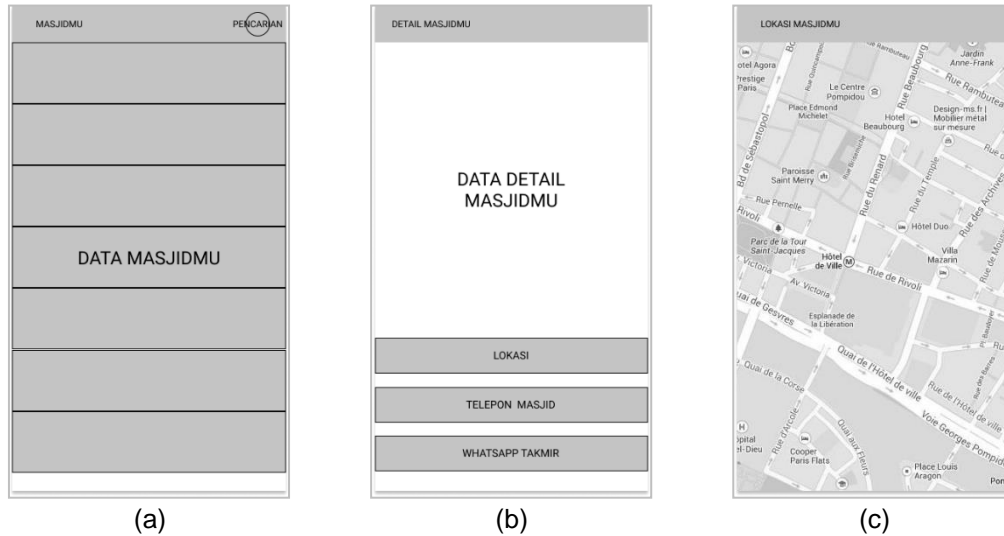
Gambar 5 di atas merupakan desain halaman Jadwal Khatib Jumat. Pada bagian tengah adalah tampilan dari Jadwal Khatib yang diambil dari format PDF. Di bagian paling bawah terdapat tombol untuk mengganti halaman PDF Jadwal Khatib Jumat.



Gambar 6. Desain Halaman Antarmuka Buklet Uswatun Hasanah

Keterangan pada Gambar 6:

- Pada bagian halaman Uswatun Hasanah terdapat gambar dan judul di setiap baris datanya.
- Tampilan antarmuka halaman artikel dari judul yang dipilih. Di sana terdapat gambar dan isi dari artikel data Uswatun Hasanah yang dipilih.



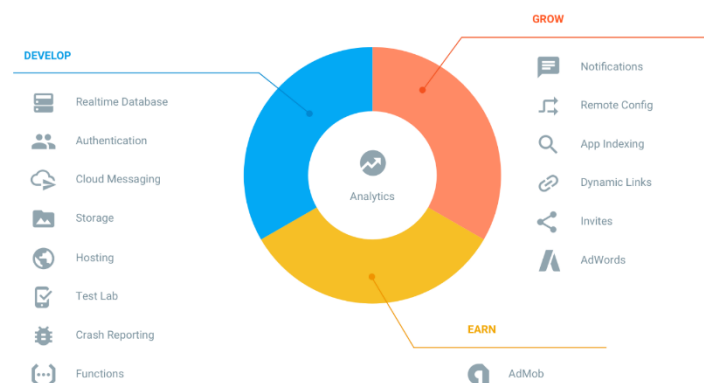
Gambar 7. Desain Halaman Antarmuka Masjid

Keterangan pada Gambar 7:

- Tampilan antarmuka halaman MasjidMu, di mana pada bagian atas terdapat tombol pencarian masjid dan pada bagian bawah terdapat data-data masjid.
- Tampilan detail dari data masjid yang dipilih. Pada halaman detail juga terdapat tiga buah tombol pilihan yaitu tombol lokasi untuk melihat lokasi masjid dan dua tombol lainnya digunakan untuk menghubungi masjid atau takmir.
- Tampilan halaman lokasi masjid terpilih.

2.4 Konstruksi

Ketika desain aplikasi telah selesai dibuat, maka selanjutnya adalah melakukan aktivitas pembangunan aplikasi dengan menulis kode-kode menggunakan alat pengembang hingga aplikasi selesai dibuat. Bahasa pemrograman yang digunakan dalam membuat aplikasi android adalah Dart yang sudah menjadi satu dengan *framework* Flutter. Kemudian basis data yang digunakan adalah *firebase realtime database* menggunakan fitur *cloud firestore*.



Gambar 8. Fitur Firebase Database [4]

Firestore Realtime Database pada Gambar 8 merupakan basis data *online* yang dapat digunakan sebagai media penyimpanan data dari aplikasi. Data disimpan dalam bentuk JSON dan dapat disinkronkan secara *realtime* ke setiap *client* yang terhubung. Setidaknya ada tiga kemampuan inti dari Firebase yang menjadi alasan digunakannya Firebase dalam pembuatan aplikasi antara lain[4].

- Realtime*, jika terdapat perubahan pada data *database*, maka seluruh *client* yang terhubung secara otomatis akan mendapatkan perubahannya dengan cepat.

2. *Offline*, yaitu aplikasi yang menggunakan fitur ini akan tetap responsif bahkan saat dalam keadaan luring. Hal ini disebabkan karena Firebase SDK (*Software Development Kit*) dapat mempertahankan data dan perubahannya pada media penyimpanan *client*. Pada saat *client* terhubung ke jaringan internet, maka Firebase SDK akan melakukan penyesuaian otomatis atas catatan perubahan data yang disimpan pada media penyimpanan *client* dengan kondisi terkini dari Firebase Server.
3. *Accessible from client devices*. Layanan ini menawarkan kemudahan untuk mengakses *firebase realtime database* secara langsung dari sebuah perangkat *mobile* atau sebuah peramban web tanpa membutuhkan server *application*.

Berikut ini adalah alat software yang digunakan untuk membuat sistem informasi atau aplikasi MakotaMu.

1. Visual code studio & Android studio, untuk menulis kode program aplikasi MakotaMu.
2. Figma, untuk membuat desain antarmuka aplikasi dan web.
3. Diagram.io dan Visual paradigm, untuk membuat desain UML.
4. Sublime text, untuk menulis kode program web admin.
5. Xampp, untuk membuat percobaan web admin sebelum di-*hosting* ke internet.
6. Corel draw, untuk membuat *icon* aplikasi.

Setelah aplikasi dibuat, selanjutnya aplikasi atau sistem akan diuji menggunakan *black-box testing* atau menguji fungsi dari setiap fitur yang ada. Hasil pengujian didokumentasikan menggunakan dokumen *User Acceptance Testing (UAT)*. Jika hasil pengujian berhasil, maka dokumen UAT akan divalidasi oleh *stakeholder* dan dapat dilanjutkan dengan aktivitas *deployment*.

2.5 Deployment

Apabila aplikasi telah divalidasi oleh *stakeholder*, maka selanjutnya adalah menyerahkan aplikasi atau sistem kepada *stakeholder*. Aplikasi MakotaMu yang sudah jadi akan dimasukkan ke *Playstore* dan web admin akan di-*hosting* ke penyedia *hosting*.

2.6 Pengujian Usability

Untuk menguji tingkat kegunaan aplikasi (*usability*) menurut sudut pandang pengguna, maka digunakan kuesioner *System Usability Scale (SUS)*[5]. SUS merupakan metode pengujian *usability* suatu sistem secara sederhana yang memiliki 10 pertanyaan dan memiliki 5 (lima) opsi jawaban pada tiap pertanyaan, mulai dari Sangat Setuju hingga Sangat Tidak Setuju. John Brooke merupakan orang yang pertama kali memperkenalkan metode ini pada tahun 1986. SUS merupakan paket pengujian *usability* yang efektif dan andal untuk digunakan pada berbagai produk dan aplikasi maupun situs web (Bangor, Kortum, & Miller, 2009)[6]. Tabel 1 di bawah ini adalah tabel kategori standar sistem berdasarkan skor nilai akhir[5].

Tabel 1. Kategori Nilai[5]

No.	Rentang Nilai	Status
1.	Skor akhir di (71-100)	<i>Acceptable</i>
2.	Skor akhir di (51-70.9)	<i>Marginal</i>
3.	Skor akhir di (0-50.9)	<i>Not Acceptable</i>

Selain kategori di atas, tingkat *usability system* dapat dikategorikan berupa *grade* beserta *rating*-nya pada Gambar 9 di bawah ini[7].

SUS Score	Grade	Adjective Rating
> 80.3	A	Excellent
68 – 80.3	B	Good
68	C	Okay
51 – 68	D	Poor
< 51	F	Awful

Gambar 9. Interpretasi Sus Score[7]

Untuk menghitung total skor, dengan cara menjumlahkan jumlah skor untuk nomor ganjil dan nomor genap, kemudian dikalikan dengan 2.5 (jumlah skor nomor ganjil + jumlah skor nomor genap) x 2.5 untuk setiap responden. Setiap skor akhir responden dijumlahkan kemudian dibagi jumlah responden, maka hasilnya akan terlihat[7]. Sebelum itu ada aturan dalam menghitung nomor ganjil dan genap (skor setiap pertanyaan adalah antara 1 sampai 5). Untuk bernomor ganjil, kurangi 1 dari skor (x-1) dan untuk bernomor genap, kurangi nilainya dari 5 (5-x)[7].

3. Hasil Penelitian dan Pembahasan

Setelah menerapkan semua aktivitas yang telah direncanakan pada metode penelitian, selanjutnya membahas hasil dari penerapan tersebut.

3.1. Implementasi *Incremental*

Hasil dari setiap tahap dalam model *incremental* telah dijelaskan sebelumnya pada sub bab metode penelitian di atas. Pada sub bab ini hanya akan menampilkan daftar *increment* yang terjadi. Tabel 2 di bawah ini adalah tabel yang berisi perubahan pada aplikasi (*increment*) selama pengembangan.

Tabel 2. *Increment Pada Aplikasi Makotamu*

No.	Tanggal rilis (2020)	Perubahan
1.	29 Juni	- Penambahan fitur MakotaMu TV (<i>link</i> ke Youtube)
2.	30 Juni	- Penambahan fitur AUM MEK (Di dalamnya ada JSM, Travel Tiga Cahaya Utama, Q-Masmu, Pasar Surya dan Surya Mart)
3.	04 Juli	- Penambahan menu Al-Bayaany

3.1 Implementasi Antarmuka

Gambar 10 berikut hasil dari penerapan desain antarmuka aplikasi yang menampilkan beberapa halaman fitur utama dari aplikasi MakotaMu.



Gambar 10 . Tampilan Antarmuka Halaman Beranda

Keterangan:

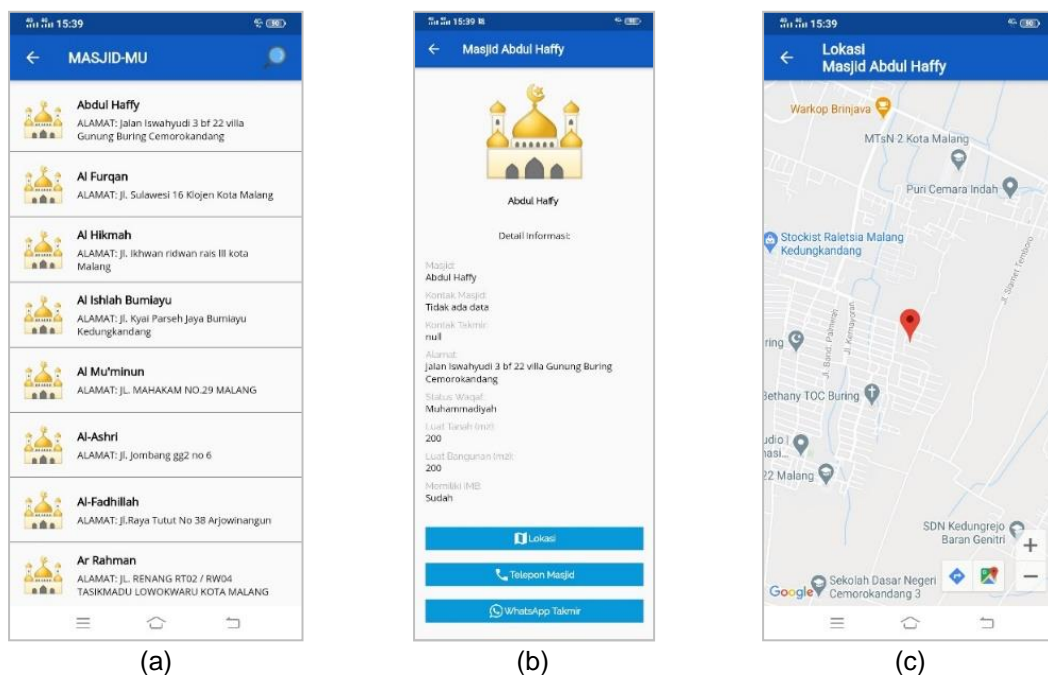
Gambar 11 halaman ini akan tampil ketika pengguna telah masuk atau mendaftar pada aplikasi sebelumnya. Pada halaman ini terdapat menu-menu pilihan yang bisa dipilih oleh pengguna.



Gambar 11. Tampilan Antarmuka Halaman Jadwal Khatib

Keterangan:

Gambar 12 halaman ini akan muncul jika pengguna memilih menu jadwal khatib pada halaman beranda. Pengguna dapat memilih halaman pada navigasi di bagian bawah layar.



(a)

(b)

(c)

Gambar 12. Tampilan antarmuka halaman masjid

Keterangan:

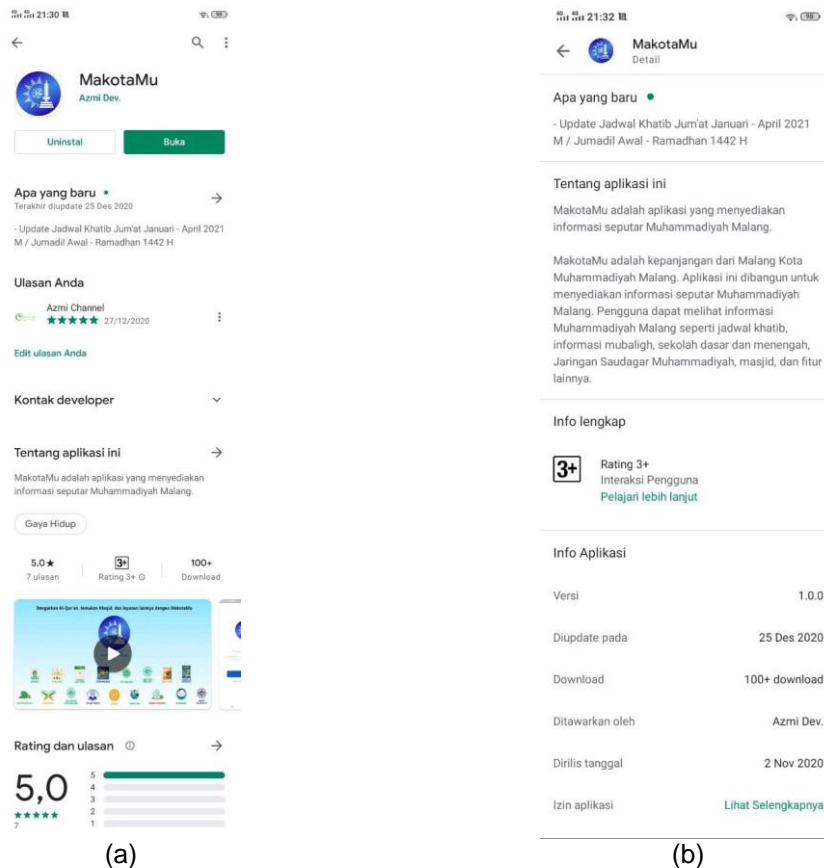
- Halaman ini akan tampil jika pengguna memilih menu MasjidMu pada halaman beranda. Pada bagian atas layar sebelah kanan terdapat tombol pencarian jika pengguna ingin melakukan pencarian data.
- Tampilan ini akan muncul jika pengguna memilih salah satu data masjid di halaman MasjidMu. Di bagian bawah layar terdapat tiga buah tombol pilihan di antaranya tombol lokasi untuk melihat lokasi masjid terpilih, tombol telepon masjid dan tombol *whatsapp* takmir.
- Halaman ini akan muncul jika pengguna memilih tombol lokasi pada halaman detail masjid.

3.2 Pengujian *Fungsional System*

Untuk menguji fungsionalitas aplikasi, dilakukan pengujian black-box dan hasilnya didokumentasikan menggunakan dokumen *User Acceptance Testing* (UAT). Hasil yang didapat adalah semua fungsi berjalan dengan baik.

3.3 Deployment

Proyek aplikasi MakotaMu yang telah selesai dibangun dan divalidasi oleh pengguna kemudian dimasukkan ke dalam platform *Playstore* agar pengguna dapat mudah memasang aplikasi pada android mereka.



(a)

(b)

Gambar 13. Aplikasi MakotaMu di Playstore

Keterangan Gambar 13:

- Tampilan aplikasi MakotaMu pada *Playstore*. Terdapat informasi bahwa aplikasi mendapat *rating* 5.0 dan aplikasi telah di-*download* +100 pengguna.
- Tampilan informasi aplikasi MakotaMu.

3.4 Hasil Pengujian *Usability System*

Tabel 3 di bawah ini adalah hasil dari pengujian *usability system* menggunakan kuesioner *System Usability Scale* (SUS) pada aplikasi MakotaMu yang diisi oleh responden melalui *google form*.

Tabel 3. Hasil kuesioner SUS

No.	Nama	Kalkulasi (Jl. Ganjil + Jl. Genap) x 2.5	Score
1.	Imam Abda'i	(20+17) x 2.5	92.5
2.	Dedy Edward	(16+15) x 2.5	77.5
3.	Yanti Aulia Faisal	(10+19) x 2.5	72.5
4.	Muhammad Alfanwahid Rovensa	(15+17) x 2.5	80
5.	Amarul Akbar	(14+14) x 2.5	70

No.	Nama	Kalkulasi (Jl. Ganjil + Jl. Genap) x 2.5	Score
6.	Maulana Aditama	(18+14) x 2.5	80
7.	Muhammad Abidin	(15+12) x 2.5	67.5
8.	Debora Rizky Mustika	(11+16) x 2.5	67.5
9.	Muhammad Iqbal	(19+14) x 2.5	82.5
10.	Nico Ryshalim	(20+16) x 2.5	90
Total Score			780

Setelah mendapatkan hasil *score* dari masing-masing responden, selanjutnya seluruh *score* dijumlahkan dan dibagi jumlah responden yaitu 10 untuk mendapatkan nilai rata-rata *score* yang dihasilkan. Total *score* yang dihasilkan adalah 780 kemudian dibagi 10. Sehingga total *score* akhir mendapatkan nilai rata-rata 78. Berdasarkan pedoman interpretasi SUS *score*, bahwa *score* 78 mendapatkan grade “B” atau “Good”[7] dan termasuk aplikasi yang “Acceptable”[5].

4. Kesimpulan

Implementasi metode pengembangan perangkat lunak *incremental* sesuai dengan kasus yang dihadapi penulis sebagai pengembang, di mana aplikasi dapat dikembangkan secara bertahap, di mulai dari memasukkan fungsi prioritas, hingga memasukkan fitur-fitur lain sebagai pelengkap. Kemudian untuk mendesain aplikasi MakotaMu penulis menggunakan metode *User Centered Design* (UCD) dengan alat dokumen *requirement checklist*, di mana dokumen ini akan diperiksa dan divalidasi langsung oleh *stakeholder*. Berdasarkan hasil pengujian *usability* menggunakan kuesioner SUS di mana aplikasi MakotaMu mendapatkan skor 78 dengan grade “B” serta termasuk dalam kategori aplikasi yang *acceptable*, berdasarkan hasil tersebut dapat disimpulkan bahwa aplikasi MakotaMu dapat diterima dan mudah digunakan.

Referensi

- [1] M. T. P. P. Muhammadiyah, *Pedoman Korps Mubaligh Muhammadiyah*. Yogyakarta, Jakarta: Muhammadiyah, 2017.
- [2] R. S. Pressman, *Rekayasa Perangkat Lunak (Pendekatan Praktisi) Buku 1*, 7th ed. Yogyakarta: Penerbit Andi, 2012.
- [3] D. S. Dewi, A. H. Brata, and L. Fanani, “Penerapan User Centered Design dalam Pembangunan Aplikasi Informasi Hostel berbasis Android,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput. e-ISSN*, vol. 2548, no. 12, p. 964X, 2018.
- [4] E. A. W. Sanadi, A. Achmad, and Dewiani, “Pemanfaatan Realtime Database di Platform Firebase Pada Aplikasi E-Tourism Kabupaten Nabire,” *J. Penelit. Enj.*, vol. 22, no. 1, pp. 20–26, 2018, doi: 10.25042/jpe.052018.04.
- [5] S. Wardani, I. G. M. Darmawiguna, and N. Sugihartini, “Usability Testing Sesuai Dengan ISO 9241-11 Pada Sistem Informasi Program Pengalaman Lapangan Universitas Pendidikan Ganesha Ditinjau Dari Pengguna Mahasiswa,” *Kumpul. Artik. Mhs. Pendidik. Tek. Inform.*, vol. 8, no. 2, p. 356, 2019, doi: 10.23887/karmapati.v8i2.18400.
- [6] R. I. Gunawan, Muh Indra Rokhmawati and N. H. Wardani, “Evaluasi dan Perbaikan Antarmuka Pengguna Menggunakan Pendekatan User Centered Design (UCD) dan Card Sorting,” vol. 1, no. 1, 2017.
- [7] H. Alathas, “Bagaimana Mengukur Kebergunaan Produk dengan System Usability Scale (SUS) Score,” *medium.com*, 2018. <https://medium.com/kelasux/bagaimana-mengukur-kebergunaan-produk-dengan-system-usability-scale-sus-score-2d6843ca780a> (accessed Jan. 10, 2021).

