

Rancang Bangun Aplikasi Chatting Berbasis Mobile Menggunakan React Native

Sabdho Cathur Kuncoro¹, Sewaka²

Teknik, Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Indonesia

E-mail: ¹sabdhocathurkuncoro@gmail.com , ²dosen00120@unpam.ac.id

Abstrak—Perkembangan teknologi informasi telah memberikan dampak yang sangat pesat terhadap perkembangan kegiatan manusia sehari-hari. Dengan adanya perkembangan teknologi saat ini, manusia dapat dengan mudah berkomunikasi dari jarak jauh satu sama lain kapanpun dan dimanapun. Dalam hal ini aplikasi dengan layanan *chatting* menjadi salah satu media untuk berkomunikasi yang banyak digunakan oleh masyarakat. Dalam pembuatan aplikasi ini, *framework* yang digunakan untuk membangun tampilan *frontend* ialah *React Native*, dan bagian *backend* menggunakan *Firebase*. Proses pengkodean dilakukan menggunakan *Visual Studio Code*. Metode yang digunakan dalam pembuatan aplikasi *chatting* ini menggunakan metode *waterfall*. Aplikasi ini dapat digunakan untuk bertukar pesan secara personal antar pengguna dan juga antar pengguna dalam satu grup. Aplikasi ini dapat digunakan untuk mengirim pesan dalam bentuk teks maupun dalam bentuk gambar. Hasil pengujian yang dilakukan menggunakan skala likert, didapatkan rata-rata total persentase sebesar 89% pengguna menyatakan bahwa aplikasi *chatting* berbasis mobile ini bisa dipergunakan sebagai media alternatif dalam bertukar pesan.

Kata Kunci: Aplikasi, *Chatting*, Komunikasi, *Framework*, *React Native*

Abstract—*The development of information technology has had a very rapid impact on the development of daily human activities. With the development of today's technology, humans can easily communicate remotely with each other anytime and anywhere. In this case the application with a chat service is one of the media for communication that is widely used by the community. In making this application, the framework used to build the frontend view is React Native, and the backend uses Firebase. The coding process is done using Visual Studio Code. The method used in making this chat application uses the waterfall method. This application can be used to exchange messages personally between users and also between users in a group. This application can be used to send messages in the form of text or in the form of images. The results of tests carried out using a Likert scale, obtained an average total percentage of 89% of users stating that this mobile-based chat application can be used as an alternative medium in exchanging messages.*

Keywords: *Application, Chatting, Communication, Framework, React Native*

1. PENDAHULUAN

Perkembangan teknologi informasi serta komunikasi memberikan dorongan yang begitu pesat terhadap perkembangan aktifitas manusia, sebagai akibatnya mempermudah manusia dalam melakukan kegiatan sehari-hari. Dengan adanya perkembangan teknologi saat ini, manusia bisa dengan mudah berkomunikasi dari jarak jauh dan tidak dibatasi oleh ruang serta jua waktu.

Chatting merupakan suatu metode komunikasi antar pengguna secara real-time, sehingga menjadikannya salah satu fasilitas media berkomunikasi yang digandrungi oleh banyak masyarakat. Beberapa aplikasi yang sudah tersebar sebelumnya di kalangan pengguna ponsel pintar dengan layanan *chatting*, antara lain Facebook Messenger, WhatsApp, Line, Telegram, Signal. Sementara itu beberapa aplikasi *chatting* buatan dalam negeri, antara lain Callind, StealthChat, dan Hi App.

Kebanyakan aplikasi ponsel pintar dengan layanan *chatting* yang telah beredar dan sudah populer digunakan oleh masyarakat Indonesia saat ini berasal dari luar negeri. Hampir tidak adanya aplikasi dengan layanan *chatting* buatan dari dalam negeri. Maka, sudah seharusnya masyarakat Indonesia untuk mulai menggunakan aplikasi dengan layanan *chatting* buatan dalam negeri. Untuk itu perlu dirancang aplikasi *chatting* yang dapat digunakan oleh berbagai kalangan dalam masyarakat dengan berbagai macam usia serta profesi dan status.

Melalui penelitian ini, diharapkan bisa membuat produk aplikasi *mobile* yang mempunyai kinerja yang baik dan beberapa pengetahuan mengenai *framework react native* untuk memanfaatkan

teknologi baru yang mudah dikelola dan dihasilkan atau dapat kita sebut *open source* bagi para pembacanya.

2. METODOLOGI PENELITIAN

Dalam melakukan upaya memperoleh data informasi maka penulis melakukan penelitian dengan menggunakan beberapa metode, diantaranya adalah:

2.1 Metode Pengumpulan Data

Untuk memperoleh data yang diperlukan dalam penyusunan skripsi ini penulis menggunakan metode pengumpulan data sebagai berikut :

a. Observasi

Pengumpulan data dengan melakukan pengamatan secara langsung terhadap objek penelitian. Mencatat beberapa hal-hal penting yang berhubungan dengan judul penelitian, sehingga memperoleh data yang lengkap dan akurat.

b. Angket

Angket atau kuesioner merupakan teknik pengumpulan data dengan cara memberikan pertanyaan atau pernyataan secara tertulis kepada responden untuk menilai produk yang telah dikembangkan (Sugiyono, 2015:216).

c. Studi Pustaka

Pengumpulan data dilakukan dengan cara membaca, mempelajari dan mencatat hal-hal penting yang berhubungan dengan masalah penelitian dari berbagai sumber, baik melalui buku-buku yang menjadi referensi maupun dari berbagai jurnal yang diperoleh melalui media internet untuk memperoleh data-data tambahan dalam rangka melengkapi penulisan ini.

2.2 Metode Pengembangan Sistem

Model *waterfall* atau metode air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengkodean, pengujian dan tahap pendukung.

1. Analisa Kebutuhan

Proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*.

2. Desain Sistem

Desain perangkat lunak adalah proses yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengodean.

3. Penulisan Kode Program

Desain harus ditranslasikan ke dalam program perangkat lunak. Hasil dari tahap ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahap desain.

4. Pengujian Program

Pengujian fokus pada perangkat lunak secara segi fungsional dan memastikan bahwa semua bagian sudah diuji. Hal ini dilakukan untuk meminimalisir kesalahan dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.

5. Penerapan Program dan Pemeliharaan

Tidak menutup kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirimkan ke *user*. Perubahan bisa terjadi karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru. Tahap pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat perangkat lunak baru.

3. ANALISA DAN PEMBAHASAN

Aplikasi yang dirancang dan dibangun dalam penelitian ini merupakan sebuah aplikasi mobile berbasis Android dengan layanan utama yaitu chat. dimana layanan chat yang dibuat ini sebagai sarana lain untuk berkomunikasi antara pengguna lain.

Proses chatting dimulai dari *user login* dengan menggunakan nomor hp sebagai autentikasi akun. Jika *user* belum terdaftar, maka sistem akan mengalihkan *user* ke halaman buat akun, *user* diharuskan mengisi data seperti foto, nama, dan bio. Jika *user* sudah terdaftar, maka *user* langsung dialihkan ke halaman utama. Setelah itu *user* bisa memilih menu chatting personal dan grup. Untuk memulai *chat* personal, *user* bisa tekan tombol tambah dan memilih pesan personal untuk memulai chat dengan *user* lain secara personal. Setelah itu sistem akan menampilkan list kontak *user* yang terdaftar di aplikasi. Setelah itu *user* dapat memilih kontak yang dituju. Sistem akan menampilkan halaman chat personal, dimana *user* dapat mengirim pesan. Untuk memulai *chat* grup, *user* bisa tekan tombol tambah dan memilih pesan grup, setelah itu *user* diharuskan mengisi beberapa data seperti foto, nama, dan deskripsi grup. Kemudian jika *user* ingin menambahkan anggota, *user* bisa membuka detail grup dan menekan tombol *add participants*.

Proses melakukan *chat* grup pada aplikasi ini dimulai ketika *user* yang sudah mendaftar pada aplikasi, memilih menu *chat* grup. Setelah itu *user* memilih *button create group* untuk membuat grup baru. *User* mengisi form tersebut yang berisikan nama grup, foto profil grup, dan deskripsi grup. Setelah berhasil membuat grup, *user* dapat memasukkan anggota ke dalam grup dengan memilih kontak yang terdaftar pada sistem. Setelah itu *user* bisa memulai *chat* dengan anggota grup lainnya dengan terlebih dahulu memilih grup yang diinginkan.

Dalam aplikasi ini dilakukan beberapa analisa kebutuhan agar sistem mampu menangani kebutuhan dari *user*, yaitu sebagai berikut:

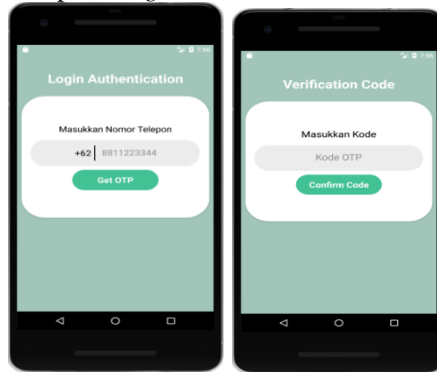
Tabel 1. Analisis dan Kebutuhan Sistem

No	Kebutuhan	Deskripsi Kebutuhan
1	Mengirim <i>Chat</i> Personal	<i>User</i> dapat mengirim <i>chat</i> kepada <i>user</i> lain, <i>chat</i> yang dikirim dapat berupa teks dan gambar
2	Melihat atau Membaca <i>Chat</i> Personal	<i>User</i> dapat melihat atau membaca <i>chat</i> personal yang dikirim oleh <i>user</i> lain
3	Membuat Grup <i>Chat</i>	Selain <i>chat</i> personal, <i>user</i> juga dapat membuat grup <i>chat</i>
4	Mengirim <i>Chat</i> Grup	<i>User</i> dapat mengirim <i>chat</i> kepada <i>user</i> lain yang berada di dalam grup
5	Melihat atau Membaca <i>Chat</i> Grup	<i>User</i> dapat melihat atau membaca <i>chat</i> grup yang dikirim oleh <i>user</i> lain
6	Melihat List Kontak <i>User</i>	<i>User</i> dapat melihat list dari kontak <i>user</i>
7	Melihat Detail <i>User</i> Lain	<i>User</i> dapat melihat detail dari <i>user</i> lain
8	Melihat Detail Grup	<i>User</i> dapat melihat detail dari grup dimana <i>user</i> merupakan anggota grup tersebut

4. IMPLEMENTASI

Implementasi yang dilakukan didasari pada beberapa perancangan yang telah dilakukan sebelumnya. Berikut ini ialah implementasi dari sistem pada aplikasi *chatting* yang penulis bangun:

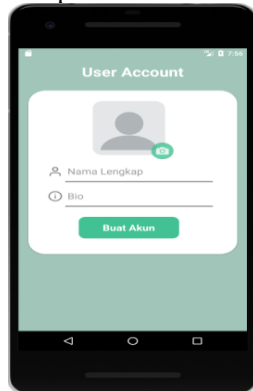
a. Tampilan *Login Authentication*



Gambar 1. Halaman *Login Authentication*

Halaman *login authentication* dipergunakan untuk autentikasi *user* agar dapat masuk ke halaman utama atau ke halaman buat akun jika *user* belum terdaftar.

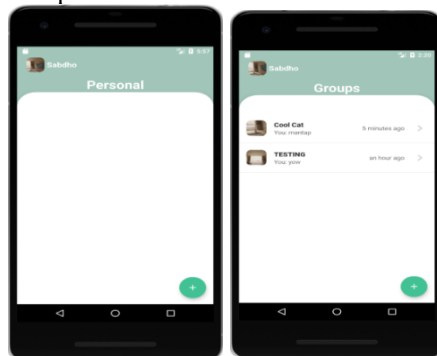
b. Tampilan Buat Akun



Gambar 2. Halaman Buat Akun

Halaman buat akun digunakan ketika *user* sudah *login* tetapi belum mempunyai akun, maka *user* diharuskan mengisi beberapa data terlebih dulu

c. Tampilan Halaman Utama



Gambar 3. Halaman Utama

Halaman utama digunakan sebagai tampilan awal ketika *user login* dan sudah terdaftar. Terdapat dua tampilan pada halaman utama, dimana terdapat daftar *chat* personal dan *chat grup*. Tampilan awal yang tertampil adalah daftar *chat* personal, untuk melihat daftar *chat* grup yaitu dengan mengusap layar ke kiri.

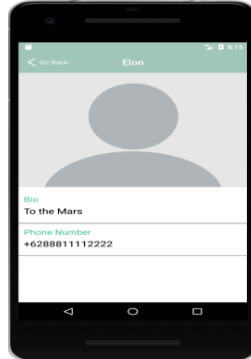
d. Tampilan Screen Chat Personal



Gambar 4. Halaman *Screen Chat* Personal

Halaman *screen chat* personal digunakan untuk mengirim pesan baik itu teks maupun gambar, jika *user* ingin mengirim pesan gambar, maka *user* diharuskan menekan tombol *icon attachment* yang ada di bawah untuk membuka galeri foto dan kemudian dikirimkan sebagai pesan gambar.

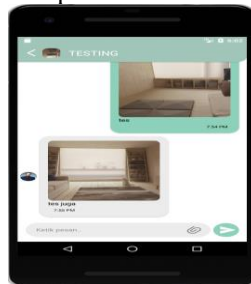
e. Tampilan Detail *User* Lain



Gambar 5. Halaman Detail *User* Lain

Halaman detail user lain akan menampilkan data diri dari *user* lain dengan menekan nama dari *user* pada halaman *screen chat* personal.

f. Tampilan *Screen Chat* Grup



Gambar 6. Halaman *Screen Chat* Grup

Halaman *screen chat* grup digunakan untuk mengirim pesan baik itu teks maupun gambar, jika *user* ingin mengirim pesan gambar, maka *user* diharuskan menekan tombol *icon attachment* yang ada di bawah untuk membuka galeri foto dan kemudian dikirimkan sebagai pesan gambar.

g. Tampilan Detail Grup



Gambar 7. Halaman Detail Grup

Halaman detail grup akan menampilkan data dari grup seperti foto, nama, deskripsi, dan anggota grup. Jika *user* ingin menambah anggota grup, maka *user* diharuskan menekan tombol *add participants*, kemudian memilih *user* yang ingin ditambahkan.

5. KESIMPULAN DAN SARAN

Berikut ini merupakan kesimpulan yang dapat ditarik dari penelitian:

- Aplikasi *chatting* berbasis mobile ini dirancang dengan UML dan dibangun menggunakan *framework react native* bertujuan untuk memudahkan *user* dalam bertukar pesan satu sama lain.
- Aplikasi *chatting* berbasis mobile ini dapat digunakan sebagai platform alternatif dalam bertukar pesan, setelah diberikan kuesioner terhadap 20 responden dan mendapatkan rata-rata total persentase sebesar 89%.

Adapun saran yang dapat penulis berikan antara lain:

- Aplikasi ini menggunakan *framework react native* yang memudahkan orang lain jika akan mengembangkan aplikasi, diharapkan dapat mengembangkan desain antarmuka aplikasi sehingga lebih menarik dan lebih memperhatikan pengalaman *user* dalam menggunakan aplikasi.
- Bagi pengembang selanjutnya, diharapkan menambahkan fitur-fitur dalam aplikasi seperti mengirim pesan video, fitur *story* tiap *user*, dan fitur *voice/video call*.

REFERENCES

- Adinta, F., & Neforawati, I. (2017). Rancang Bangun Aplikasi Chatting Berbasis Web Menggunakan Docker. *JOISIE - Journal Of Information System And Informatics Engineering*.
- Jaya, I. W., & Giri, G. M. (2019). Analisis dan Perancangan Aplikasi Chatting (DChat) Pada Rumah Sakit Umum Permata Hati Berbasis Mobile. *Jurnal Elektronik Ilmu Komputer Udayana*.
- Khairil, A., & Hayati, R. S. (2020). Rancang Bangun Aplikasi Chatting Keluarga Menggunakan Fitur Device Location Berbasis Android. *Jurnal Mahasiswa Fakultas Teknik dan Ilmu Komputer*.



- Lilipaly, N. K. (2016). Rancang Bangun Aplikasi Schoolish Berbasis Mobile Menggunakan Framework React Native. *Jurnal Skripsi Teknik Informatika dan Komputer, Politeknik Negeri Jakarta*.
- Mahir, H. M. (2019). Pembangunan Aplikasi Pemantauan Kegiatan Siswa Berbasis Mobile Menggunakan Framework React Native (Studi Kasus: Pemantauan Kegiatan Siswa di MI Cahaya Cimahi).
- Pressman, R. S., & Maxim, B. R. (2014). *Software Engineering : A Practitioner's Approach*. New York: McGraw-Hill.
- Sugiyono. (2013). *Metode Penelitian Bisnis*. Bandung: Alfabeta.
- Sukamto, R. A., & Shalahuddin, M. (2013). *Rekayasa Perangkat Lunak Terstruktur Dan Berorientasi Objek*. Bandung: Informatika.
- Sutabri, T. (2012). *Analisis Sistem Informasi*. Yogyakarta: Andi.
- Wilianti, A. (2021). Perancangan Perangkat Lunak Untuk Menggambar Diagram Berbasis Android. *Syntax Literate: Jurnal Ilmiah Indonesia*, 4.