

APLIKASI KONSULTASI DOKTER-PASIEN SECARA ONLINE BERBASIS ANDROID MENGGUNAKAN REACT NATIVE

Nicolas Sanjaya^{1*}, Mochamad Wisuda Sardjono²

¹Program Studi Teknik Informatika, Universitas Gunadarma

²Program Studi Sistem Informasi, Universitas Gunadarma

Jalan Margonda Raya No. 100, Depok

Email: ^{1*}nicolas.sanjaya701@gmail.com, ²moch_wisuda@staff.gunadarma.ac.id

Abstract

The use of communication technology has penetrated into every field, including those related to medical consulting services. Smartphone devices not only function as a verbal communication tool, but can be used for online communication. With the development of computer programming languages, smartphone devices can be used more widely to serve consultations about health. Javascript is a programming language that is widely used in developing mobile applications. The React Native framework is used to design mobile applications so that they are easy to use via smartphone devices without going through a web application. In this study UML is used as an application design method and firebase as a data storage medium. The trials were carried out using BlackBox Testing on 4 smartphone devices based on the Android operating system with different specifications. In addition, trials were also carried out on every function in the Doctor-Patient Consultation application involving 25 respondents by providing an assessment. The test results show that the use of the React Native framework maximizes the design of Android-based mobile applications. Of the 4 smartphone devices, all functions in the application went well, apart from that, 25 respondents who gave an assessment of more than 65% stated that they agreed and strongly agreed with each statement regarding the application.

Keywords: Android, Framework, JavaScript, Native, React

PENDAHULUAN

Sejak menyebarnya penyakit COVID-19, setiap kegiatan masyarakat memiliki keterbatasan untuk berinteraksi. Pengetatan kegiatan berkumpul pun dilakukan oleh pemerintah untuk menghentikan penyebaran penyakit tersebut. Dengan kondisi tersebut mendasari pemanfaatan teknologi komunikasi dan informasi secara maksimal. Sehingga beberapa pendapat pelaku kesehatan mengharuskan adanya solusi yang dapat diberikan untuk memberikan layanan konsultasi kesehatan. Ketika seseorang merasa sedang sakit terkadang akan mencari obat dengan cara *browsing* di internet, hal tersebut tidak dianjurkan karena dapat menjadi salah dalam menentukan obat pilihan lebih baik menggunakan telemedika (Karisma, 2021). Konsultasi dokter online dapat membawa keuntungan bagi masyarakat mengetahui kondisi kesehatan tanpa perlu berkunjung ke rumah sakit dan bertemu dokter secara langsung (Nareza, 2021).

Membuat aplikasi *mobile* seringkali cenderung menjadi pekerjaan yang sulit bagi seorang pengembang aplikasi karena harus membuat satu aplikasi untuk platform Android dan iOS yang membutuhkan pengetahuan tentang dua domain yang berbeda, pengetahuan tentang IDE (Studio Android, bahasa pemrograman: Kotlin atau Java) dan untuk aplikasi seluler iOS (bahasa pemrograman: swift atau objektif C dan IDE: Xcode) diperlukan (Kaushik, Gupta, & Gupta, 2018). JavaScript diperkenalkan pada tahun 1995 sebagai cara untuk menambahkan program ke halaman web di *browser* Netscape Navigator (Intern, 2020). Bahasa ini telah diadopsi oleh semua *browser web* grafis utama lainnya. Itu telah memungkinkan aplikasi web modern untuk berinteraksi secara langsung, tanpa melakukan pemuatan ulang halaman untuk setiap tindakan. Tapi itu juga digunakan di situs web yang lebih tradisional untuk menyediakan berbagai bentuk interaktivitas dan kepintaran (Haverbeke, 2014) (Paul & Nalwaya, 2019) (Kaushik, Gupta, & Gupta, 2018). React muncul karena pengembang aplikasi dihadapkan pada masalah yang signifikan: bagaimana membangun aplikasi besar yang datanya sering berubah. Masalah ini

terjadi di hampir semua aplikasi dunia nyata dan React diciptakan dari bawah ke atas untuk menyelesaikannya. Seperti yang diketahui, *framework* populer adalah MVC atau model-view-wildcard (MV*), tetapi ada hal yang perlu diperhatikan dan diulangi: React bukanlah kerangka kerja MV*.

React hanyalah *library* untuk membuat *User Interface* (UI) yang dapat disusun untuk komponen UI dengan data yang berubah seiring waktu (Sulthon, 2020). Tidak seperti *framework* JavaScript yang populer, React tidak menggunakan *template* atau arahan *Hypertext Markup Language* (HTML) (Eisenman, 2016). React membangun UI dengan memecah UI menjadi banyak komponen. React menggunakan fitur lengkap dari bahasa pemrograman untuk membangun dan merender tampilan (Paul & Nalwaya, 2019). Penelitian mengenai *framework* React Native telah banyak dilakukan, dengan memanfaatkan perpaduan dua *framework*, Adobe Phonegap (Phonegap) dan React Native diterapkan pada pembuatan aplikasi hibrid *mobile*. Studi kasus pada Aplikasi Jadwal dan Nilai pada Kota Malang (Wijanarko & Aji, 2018). Pemanfaatan *framework* React Native berfungsi dengan baik dengan berbeda perangkat dan sistem operasi: pengembangan aplikasi pemesanan minuman kopi melalui *smartphone* Android dan iOS (Malahella, Arwani, & Tibyani, 2020), rancang bangun aplikasi *mobile* pada platform iOS dan Android (Kapantow, Lumenta, & Sambul, 2020), pengembangan aplikasi *mobile* untuk *e-ticketing* pertandingan sepak bola pada *smartphone* berbasis Android 5 (Armi & Budayawan, 2021).

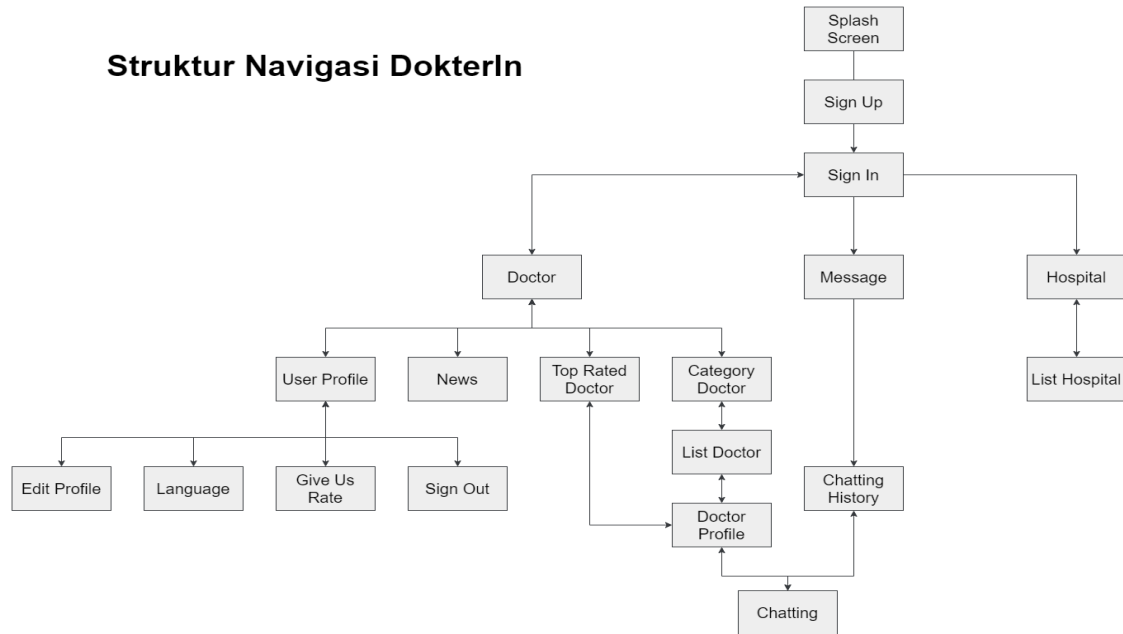
Dalam penelitian ini mengusulkan untuk mengetahui kinerja *framework* React Native karena *framework* tersebut sering digunakan pada pembuatan Aplikasi berbasis Android. Aplikasi konsultasi dokter-pasien ini diterapkan pada empat perangkat komunikasi dengan berbasis Android 7, 10 dan 11.

METODE PENELITIAN

Metode penelitian yang diusulkan oleh penulis adalah menggunakan UML. Untuk memudahkan dalam menjelaskan tahapan-tahapan yang dilakukan pada penelitian ini maka penulis menjelaskan pada langkah Struktur Navigasi, Use Case Diagram dan Rancangan Interface yang digunakan dalam aplikasi.

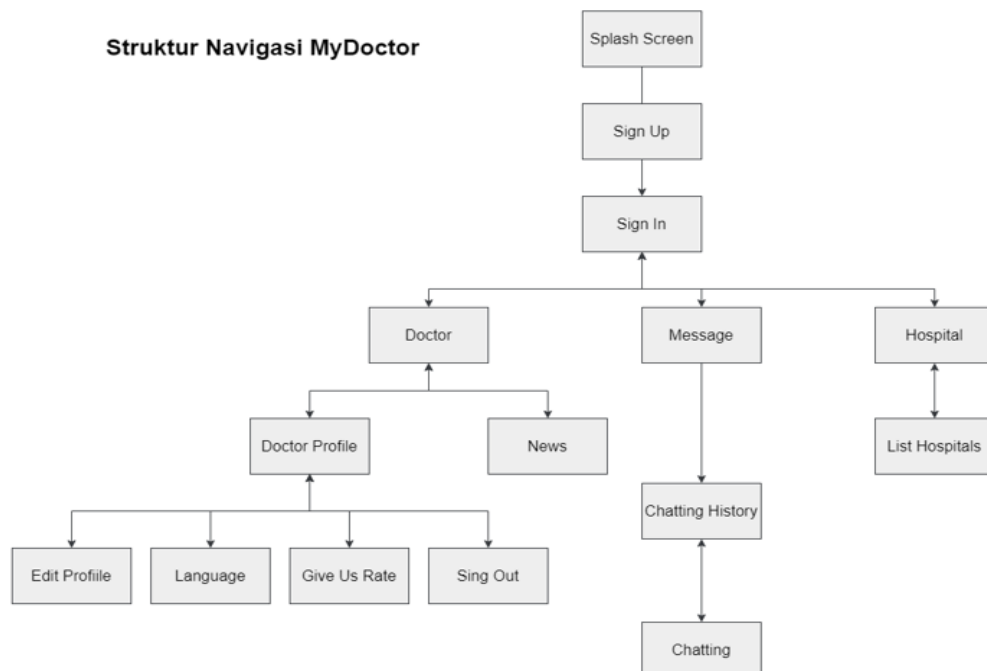
Struktur Navigasi

Langkah awal dalam perancangan aplikasi ini adalah membuat struktur navigasi. Langkah ini merupakan langkah yang penting untuk dilakukan. Struktur navigasi yang digunakan adalah struktur navigasi campuran. struktur navigasi campuran pada aplikasi ini merupakan gabungan dari struktur navigasi linear, non linear dan hirarki. Struktur navigasi DokterIn dijelaskan langkah yang dilakukan oleh Dokter untuk melakukan registrasi atau pendaftaran, sehingga pasien akan mengetahui Dokter yang aktif pada aplikasi. Dalam proses registrasi ini, Dokter mengisi data pribadi diantaranya nomor registrasi profesi dokter, alur tersebut ditunjukkan pada Gambar 1. Berdasarkan Gambar 1, saat kedua aplikasi dijalankan tampilan yang pertama kali muncul adalah Splash Screen yang akan menampilkan logo dan nama dari aplikasi dalam waktu tertentu. Selanjutnya pengguna yang akan menggunakan aplikasi ini harus melakukan daftar akun terlebih dahulu kemudian setelah itu dapat login. Jika pengguna sudah login maka ditampilkan halaman utama yaitu Doctor. Di dalam aplikasi DokterIn terdapat berbagai macam fitur, yaitu *User Profile*, *News*, *Top Rated Doctor* dan *Category Doctor*.



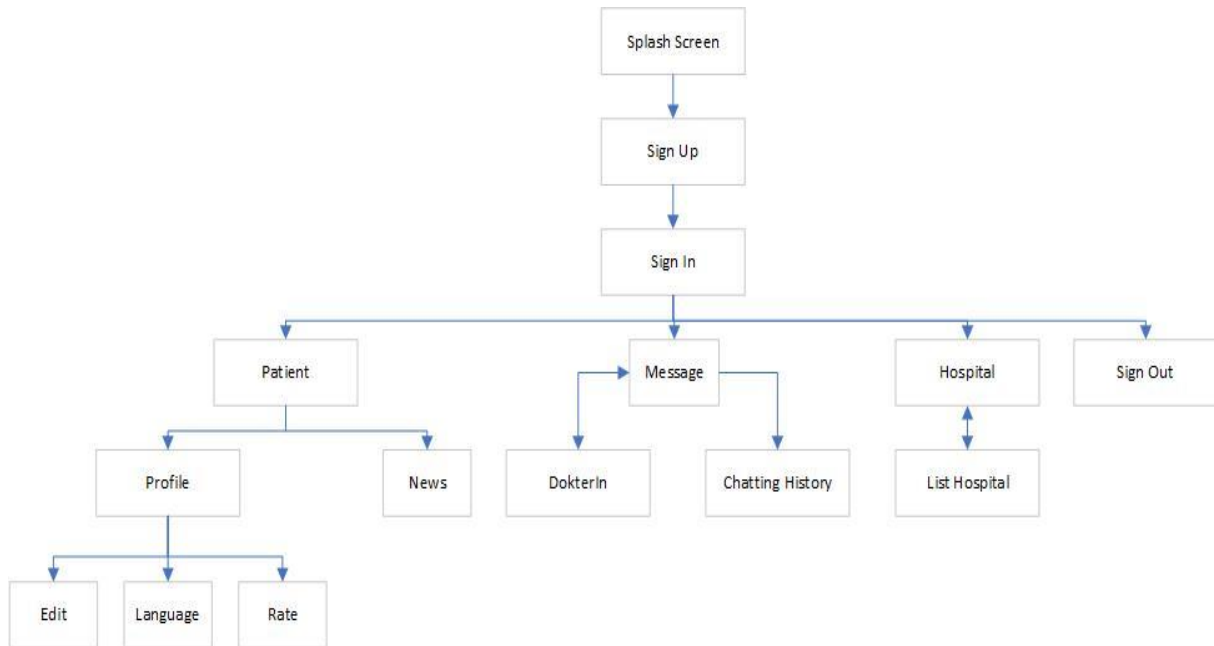
Gambar 1. Struktur Navigasi Aplikasi DokterIn

Pada tampilan Doctor pada aplikasi MyDoctor terdapat 2 fitur, yaitu Doctor Profile dan News. Jika pengguna memilih menu category doctor, maka pengguna akan memilih dokter sesuai dengan pilihan yang ada, dapat dilihat pada Gambar 2. Kemudian jika sudah memilih dokter, pengguna akan memilih profile dari dokter tersebut, lalu dapat berkonsultasi dengan dokter tersebut. Pada tampilan Doctor juga terdapat menu User Profile, dan pengguna dapat mengubah data diri pada menu Edit Profile. Lalu terdapat Language, Give Us Rate dan Sign Out untuk keluar dari aplikasi. Pada tampilan Message terdapat *history chat* antara pengguna dengan dokter. Selanjutnya pada halaman terakhir yaitu Hospital dimana terdapat list hospital.



Gambar 2. Struktur Navigasi Aplikasi MyDoctor

Selain struktur navigasi DokterIn dan MyDoctor, dibuat juga struktur navigasi Pasien yang dapat diintegrasikan seluruh alur data dalam suatu sistem dan instruksi yang dilakukan oleh pengguna, ditunjukkan pada Gambar 3. Dijelaskan bahwa pengguna harus melakukan Sign Up terlebih dahulu untuk dapat masuk kedalam aplikasi. Jika sudah melakukan Sign Up, maka pengguna dapat masuk kedalam aplikasi dengan cara Sign In menggunakan email dan password yang sudah dibuat sebelumnya. Pengguna akan masuk kehalaman Home Patient, kemudian pengguna dapat memilih Message dan memilih DoctorIn yang diinginkan.



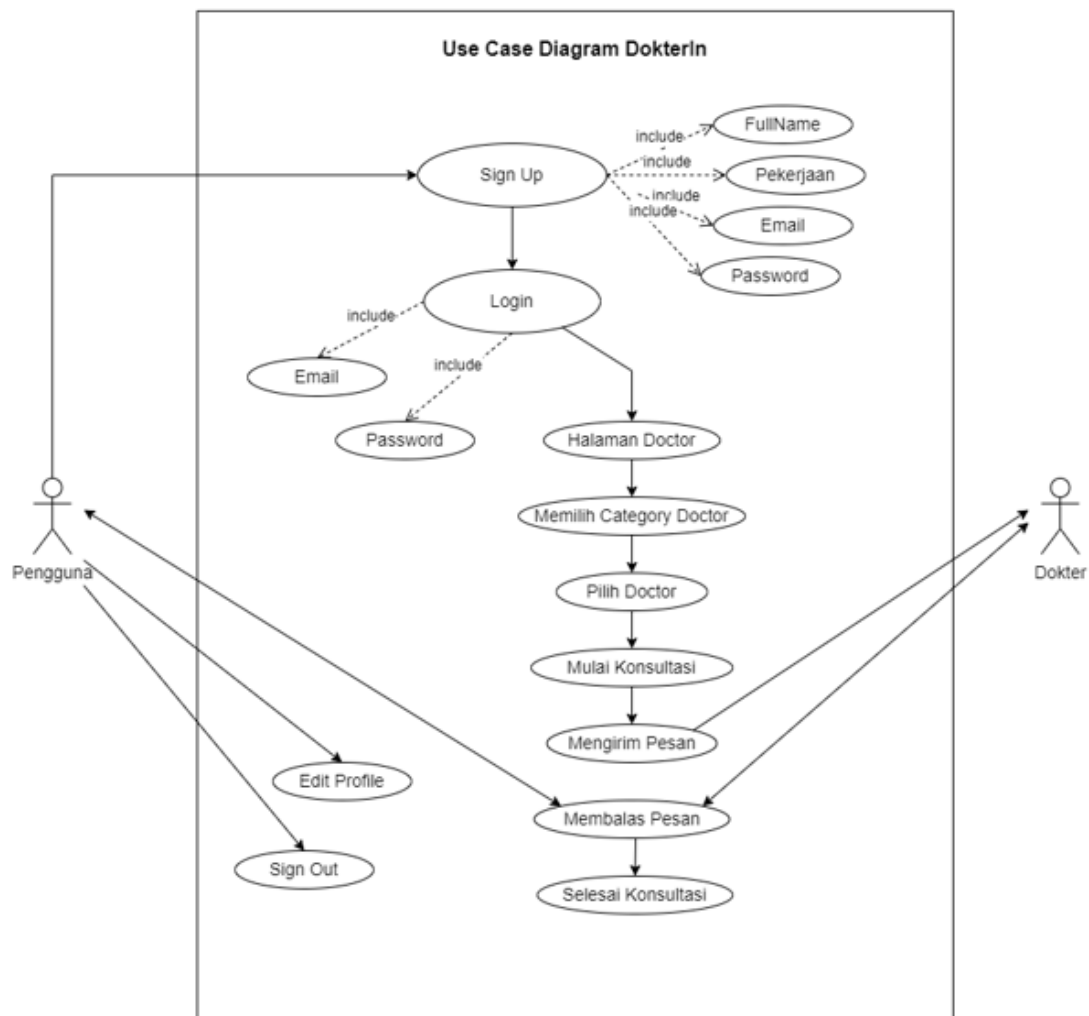
Gambar 3. Struktur Navigasi Pasien

Selanjutnya pengguna akan memilih DoctorIn yang tersedia pada bagian kategori Message. Setelah memilih dokter yang dipilih maka dapat memulai konsultasi dengan menekan “Start Consultation” dan akan masuk ke halaman Chatting. Pengguna dapat langsung mengirim pesan ke dokter yang telah dipilih, kemudian dokter akan membalas pesan dari pengguna pada menu messages. Setelah melakukan konsultasi dengan dokter, pengguna juga dapat melakukan Edit Profile dan Sign Out untuk dapat keluar dari aplikasi DokterIn.

Use Case Diagram

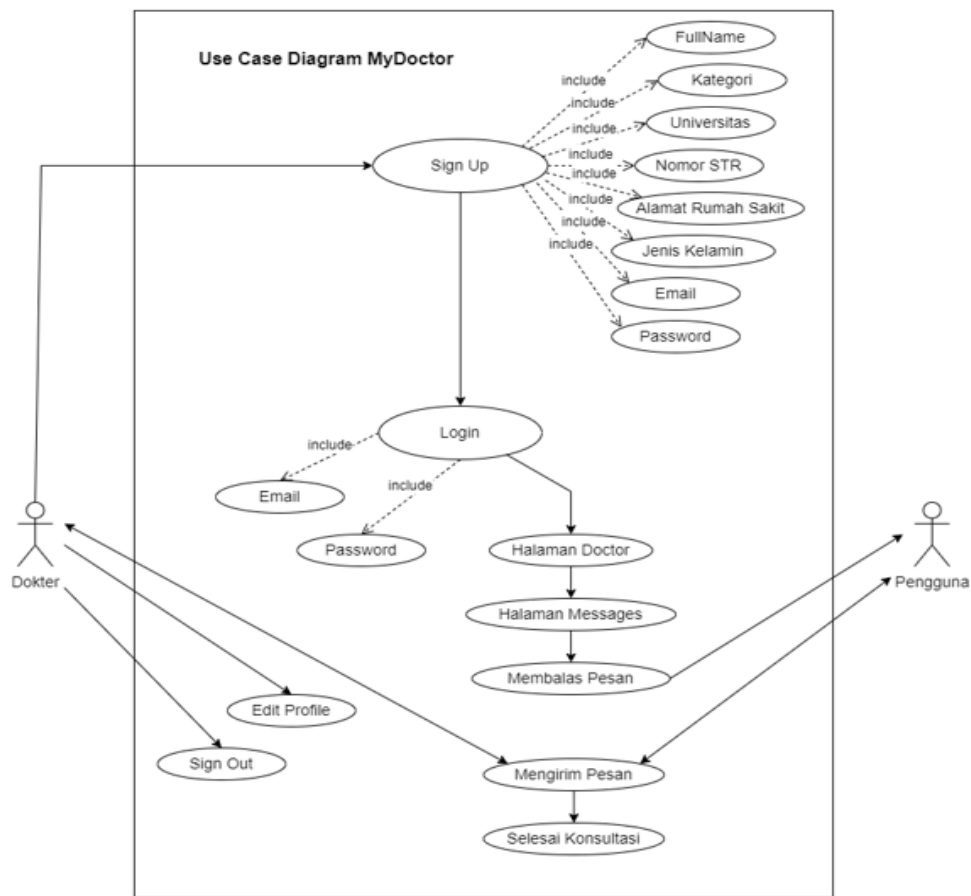
Pada perancangan ini, dibuat use case diagram yang dapat diintegrasikan seluruh alur data dalam suatu sistem dan instruksi yang dilakukan oleh pengguna, Use Case Diagram pada aplikasi ini ditunjukkan pada 4 dan 5.

Pada Gambar 4 dijelaskan bahwa pengguna harus melakukan Sign Up terlebih dahulu untuk dapat masuk kedalam aplikasi. Jika sudah melakukan Sign Up, maka pengguna dapat masuk kedalam aplikasi dengan cara Sign In menggunakan email dan password yang sudah dibuat sebelumnya. Pengguna akan masuk kehalaman Home Page atau Halaman Doctor, kemudian pengguna dapat memilih Category Doctor yang diinginkan. Selanjutnya pengguna akan memilih doctor yang tersedia pada bagian Category Doctor. Setelah memilih doctor yang dipilih maka dapat memulai konsultasi dengan menekan “Start Consultation” dan akan masuk ke halaman Chatting. Pengguna dapat langsung mengirim pesan ke dokter yang telah dipilih, kemudian dokter akan membalas pesan dari pengguna pada menu messages. Setelah melakukan konsultasi dengan dokter, pengguna juga dapat melakukan Edit Profile dan Sign Out untuk dapat keluar dari aplikasi DokterIn.



Gambar 4. Use Case Diagram Aplikasi DokterIn

Pada Gambar 5. dijelaskan bahwa pengguna harus melakukan Sign Up terlebih dahulu untuk dapat masuk kedalam aplikasi. Jika sudah melakukan Sign Up, maka pengguna dapat masuk kedalam aplikasi dengan cara Sign In menggunakan email dan password yang sudah dibuat sebelumnya. Dokter akan masuk kehalaman Home Page atau Halaman Doctor, lalu dokter dapat memilih halaman Messages untuk membalas pesan dari pengguna. Kemudian dokter juga dapat melakukan Edit Profile dan Sign Out untuk keluar dari aplikasi.



Gambar 5. Use Case Diagram Aplikasi MyDoctor

HASIL DAN PEMBAHASAN

Pada halaman Daftar Akun pengguna diharuskan mengisi nama lengkap, pekerjaan, email dan password kemudian dapat klik tombol continue. Pada Daftar Akun Dokter diharuskan mengisi nama lengkap, kategori, universitas, nomor STR, alamat rumah sakit, jenis kelamin, email dan password. Tampilan Daftar Akun dapat dilihat pada Gambar 6.

(a)

(b)

Gambar 6. (a) Tampilan Daftar Akun Pasien (b) Tampilan Daftar Akun Dokter

Pada halaman Login pengguna dan dokter akan memasukkan data email dan password yang telah dibuat pada halaman daftar akun. Jika pengguna dan dokter belum mempunyai akun maka dapat membuat akun terlebih dahulu pada bagian “create new account”. Tampilan Login dapat dilihat pada Gambar 7.

(a)

(b)

Gambar 7. (a) Tampilan Login Akun Pasien (b) Tampilan Login Akun Dokter

Pembuatan halaman ini terdapat pada fule Chatting/index.js. Pada halaman ini pengguna dapat menuliskan pesan yang ingin disampaikan kepada dokter dan dapat melihat balasan dari dokter maupun sebaliknya. Tampilan Chatting dapat dilihat pada Gambar 8.

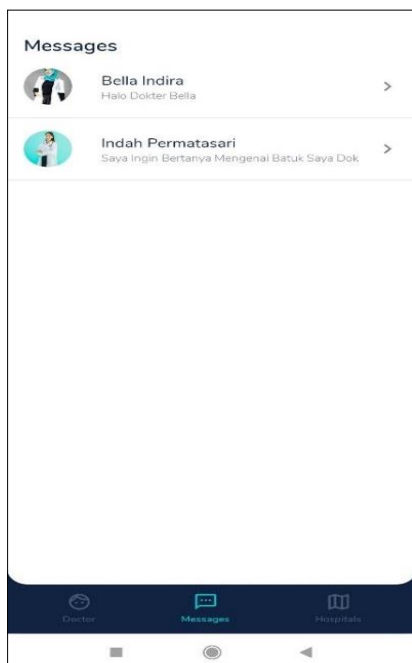


(a)

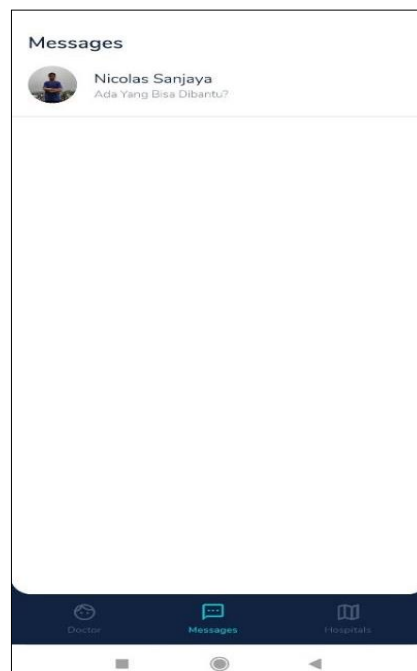


(b)

Gambar 8. (a) Tampilan Chat Dokter (b) Tampilan Chat Pasien



(a)



(b)

Gambar 9. (a) Tampilan Message Dokter (b) Tampilan Message Pasien

Pada Gambar 9 ditampilkan halaman Chatting, pembuatan halaman ini terdapat pada file Chatting/index.js. Pada halaman ini pengguna dapat menuliskan pesan yang ingin disampaikan kepada dokter dan dapat melihat balasan dari dokter maupun sebaliknya

Hasil uji coba yang dilakukan pada penelitian yang dilakukan dengan pemanfaatan *framework vue js* dan *Quasar* tidak disertakan pengujian terhadap beberapa perangkat *smartphone* sehingga tidak diketahui kinerja dari *framework* tersebut [25]. Pengembangan aplikasi konsultasi dokter hewan menggunakan bahasa pemrograman Kotlin dilakukan dengan menguji pada 2 perangkat *smartphone* dengan hasil yang menyatakan aplikasi tersebut berjalan dengan baik [26]. Uji coba *framework react native* dilakukan pada aplikasi Edukasi Bebahasa Jepang berbasis Android dengan hasil aplikasi berhasil diimplementasikan [23]. Framework React Native juga digunakan pada aplikasi *mobile* jasa layanan penyeberangan kapal laut. Dengan mendapatkan hasil uji coba pada aspek *functionality suitability* sebesar 99,8%, aspek *usability* 94,6% dan aspek *compatibility* 100% [24].

Proses uji coba aplikasi menggunakan metode *black box testing* pada perangkat *smartphone* Redmi Note 9 Pro, Redmi 4x, Poco X3 Pro dan Redmi 8 Pro. Hasil dari uji coba dapat dilihat pada Tabel 1.

Tabel 1. Hasil Uji Coba Aplikasi menggunakan Black Box pada perangkat *smartphone* Redmi Note 9 Pro, Redmi 4x, Poco X3 Pro dan Redmi 8 Pro.

No	Fungsi	Input	Goal	Output	Hasil
1	Tampilan Splash Screen	Pengguna membuka aplikasi	Menampilkan halaman Splash Screen berdurasi 3 detik	Tampilan Splash Screen menampilkan hasil yang sesuai	Sukses
2	Tampilan Get Started	Pengguna masuk pada halaman get started	Menampilkan 2 button yaitu get started dan sign in	Halaman get started ditampilkan	Sukses
3	Tampilan Daftar Akun	Pengguna mengisi data dengan benar pada bagian daftar akun	Menampilkan 4 input text dan 1 button continue	Halaman daftar akun ditampilkan	Sukses
4	Tampilan Upload Photo	Pengguna mengunggah photo yang dipilih	Menampilkan 1 photo, nama pengguna, pekerjaan, 1 button upload and continue dan 1 link skip for this	Halaman upload photo ditampilkan	Sukses
5	Tampilan Login	Pengguna mengisi email dan password sesuai yang sudah dibuat	Menampilkan 2 input text dan 1 button sign in	Halaman sign in ditampilkan	Sukses
6	Tampilan Home Doctor	Pengguna masuk ke bagian home doctor setelah login	Menampilkan profile pengguna seperti nama, pekerjaan dan photo profile. Terdapat juga category doctor, top rated doctor dan news	Halaman home doctor ditampilkan sesuai	Sukses
7	Tampilan User Profile	Pengguna klik pada bagian profil di home doctor	Menampilkan photo pengguna, nama, pekerjaan, edit profile, language, give us rate dan	Halaman user profile ditampilkan	Sukses

8	Tampilan Edit Profile	Pengguna klik pada bagian edit profile di user profile	sign out Menampilkan photo pengguna, 3 input text, 1 static text dan 1 button save profile	Halaman edit profile ditampilkan	Sukses
9	Tampilan Category Pilih Dokter	Pengguna memilih category berdasarkan yang diinginkan	Menampilkan list dokter yang tersedia	Halaman category pilih dokter ditampilkan sesuai	Sukses
10	Tampilan Doctor Profile	Pengguna dapat melihat data dari dokter yang dipilih	Menampilkan data profile dokter dan 1 button start consultation	Halaman doctor profile ditampilkan sesuai	Sukses
11	Tampilan Chatting	Pengguna masuk setelah klik start consultation pada halaman doctor profile	Menampilkan photo, nama, dan kategori dokter yang dipilih. Terdapat input text dan send.	Halaman chatting ditampilkan sesuai	Sukses
12	Tampilan Messages	Pengguna melihat riwayat chatting sebelumnya	Menampilkan chatting terakhir antara dokter dan pengguna	Halaman messages ditampilkan sesuai	Sukses
13	Tampilan Hospitals	Pengguna dapat melihat rumah sakit yang tersedia	Menampilkan photo, nama dan alamat dari rumah sakit	Halaman hospitals ditampilkan sesuai	Sukses

Penulis melakukan uji coba terhadap aplikasi yang telah dibuat pada beberapa *smartphone* android dengan berbagai spesifikasi yang berbeda dan 100% dinyatakan tersebut kompatibel terhadap semua perangkat. Berikut adalah tabel uji coba pada perangkat *smartphone* berbasis android, seperti pada Tabel 2.

Tabel 2. Tabel Uji Coba Perangkat

No	Jenis Smartphone	Spesifikasi (OS, Layar, Processor)	Instalasi	Kualitas Gambar	Kualitas Layout	Error
1.	Redmi Note 9 Pro	Android 11, 6.67", Qualcomm Snapdragon 720G	Berhasil	Baik	Baik	Tidak Ada
2.	Redmi 4x	Android 7.1.2, 5.0", Qualcomm Snapdragon 435	Berhasil	Baik	Baik	Tidak Ada
3.	Poco X3 Pro	Android 11, 6.67", Qualcomm Snapdragon 860	Berhasil	Baik	Baik	Tidak Ada
4.	Redmi 8 Pro	Android 10, 6.53", Mediatek Helio G90T	Berhasil	Baik	Baik	Tidak ada

Dapat dilihat pada Tabel 3 hasil pengujian aplikasi melalui kuesioner terhadap 25 orang responden dengan mengisi 10 pertanyaan. Terdapat lebih dari 70% responden menyatakan Setuju dan Sangat Setuju. Dari hasil tersebut menunjukkan aplikasi ini sudah dapat diterima dengan baik oleh 25 responden, seperti dari tampilan, bahasa dan fitur yang jelas dan mudah untuk dimengerti pengguna.

Tabel 3. Hasil dari pengujian aplikasi dengan melibatkan 25 responden yang berpartisipasi dalam kuesioner.

No	Pertanyaan	Tidak Setuju	Netral	Setuju	Sangat Setuju
1.	Apakah tampilan aplikasi DokterIn sudah sesuai dengan harapan pengguna?	1/25 * 100 = 4%	5/25 * 100 = 20%	15/25 * 100 = 60%	4/25 * 100 = 16%
2.	Apakah perpaduan warna pada aplikasi DokterIn menarik?	0	6/25 * 100 = 24%	14/25 * 100 = 56%	5/25 * 100 = 20%
3.	Apakah huruf dalam aplikasi DokterIn mudah dibaca?	0	6/25 * 100 = 24%	14/25 * 100 = 56%	5/25 * 100 = 20%
4.	Apakah aplikasi DokterIn memiliki fitur yang jelas dan mudah dalam pemakaiannya?	0	7/25 * 100 = 28%	15/25 * 100 = 60%	3/25 * 100 = 12%
5.	Apakah pengguna tidak merasa kesulitan dalam menggunakan aplikasi DokterIn?	1/25 * 100 = 4%	7/25 * 100 = 28%	14/25 * 100 = 56%	3/25 * 100 = 12%
6.	Apakah tampilan aplikasi DokterIn menarik secara keseluruhan?	0	6/25 * 100 = 24%	15/25 * 100 = 60%	4/25 * 100 = 16%
7.	Apakah pengguna mudah dalam melakukan konsultasi dokter pada aplikasi DokterIn?	1/25 * 100 = 4%	7/25 * 100 = 28%	15/25 * 100 = 60%	2/25 * 100 = 8%
8.	Apakah pengguna merasa senang dalam menggunakan aplikasi DokterIn?	0	8/25 * 100 = 32%	14/25 * 100 = 56%	3/25 * 100 = 12%
9.	Apakah pengguna nyaman dalam menggunakan aplikasi DokterIn?	1/25 * 100 = 4%	7/25 * 100 = 28%	15/25 * 100 = 60%	2/25 * 100 = 8%
10.	Apakah tampilan aplikasi sudah terorganisir dengan baik?	1/25 * 100 = 4%	6/25 * 100 = 24%	16/25 * 100 = 64%	2/25 * 100 = 8%

KESIMPULAN

Pemanfaatan framework React Native sangat membantu pengembang aplikasi untuk mengembangkan aplikasi *mobile* secara mudah dan aplikatif yang mudah untuk dipelajari. Dari pengujian aplikasi pada empat perangkat *smartphone* berjalan dengan baik, dilihat dari instalasi, kualitas tampilan dan gambar. Pengembang sangat terbantu dengan adanya framework ini, dengan perbedaan sistem operasi *smartphone* aplikasi tetap dapat dijalankan. Dari hasil ujicoba dalam penelitian ini pun 70% dari 25 responden pengguna menyatakan pemanfaatan *framework React Native* sangat berguna pada aplikasi ini terhadap perangkat berbasis Android.

DAFTAR PUSTAKA

- Armi, H., & Budayawan, K. (2021). Pengembangan Aplikasi Mobile Menggunakan Javascript React Native untuk Ticketing Pertandingan Sepakbola. *Jurnal Vocational Teknik Elektronika dan Informatika*, 9(1), 152-161.
- Eisenman, B. (2016). *Learning React Native: Building Native Mobile Apps With JavaScript*. California: O'Reilly Media.
- Haverbeke, M. (2014). *Eloquent JavaScript A Modern Introduction to Programming*. MIT - Creative Commons.
- Intern, D. (2020, 12 2). *Apa itu JavaScript? Fungsi dan Contohnya*. Retrieved Januari 26, 2022, from <https://www.dicoding.com/blog/apa-itu-javascript-fungsi-dan-contohnya>
- Kapantow, J. P., Lumenta, A. S., & Sambul, A. M. (2020). Rancang Bangun Aplikasi Bakudapa Manado. *Jurnal Teknik Elektro dan Komputer*, 9(3), 155-162.

- Karisma, N. (2021, Januari 28). *Konsultasi Dokter Online: Mungkinkah?* Retrieved Januari 26, 2022, from <https://lifepack.id/konsultasi-dokter-online-mungkinkah/>
- Karnadi, Jimmie, Virgana, W., & Al-Amin. (2021). PENGEMBANGAN APLIKASI MOBILE MENGGUNAKAN TEKNOLOGI WEB STUDI KASUS KONSULTASI DOKTER. *Jurnal Digital Teknologi Informasi Universitas Muhammadiyah*, 4(2), 73-80.
- Kaushik, V., Gupta, K., & Gupta, D. (2018). React Native Application Development. *Proceedings of 4th International Conference on Cyber Security (ICCS)*. Jaipur (Rajasthan), India.
- Malahella, A. H., Arwani, I., & Tibyani. (2020). Pemanfaatan Framework React Native dalam Pengembangan Aplikasi Pemesanan Minuman Kopi pada Kedai Bycoffee. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 4(9), 3178-3184.
- Nareza, d. M. (2021, Maret 01). *Memahami Konsultasi Dokter Online*. Retrieved Januari 26, 2022, from <https://www.alodokter.com/konsultasi-dokter-online>
- Paul, A., & Nalwaya, A. (2019). *React Native for Mobile Development: Harness the Power of React Native to Create Stunning iOS and Android Application 2nd Edition*. New York: APress Media.
- Santoso, A. B., Prasetyo, A. B., & Windasari, I. P. (2022). PERANCANGAN APLIKASI ANDROID KONSULTASI KESEHATAN MENGGUNAKAN REACT NATIVE. *Jurnal Ilmu Teknik dan Komputer*, 6(1), 76-85.
- Sulthon, A. (2020, Juli 16). *React Native : Apa itu, Penjelasan, 10+ Kelebihan dan Kekurangan*,. Retrieved Januari 26, 2022, from <https://www.domainesia.com/tips/react-native/>
- Wijanarko, D., & Aji, R. F. (2018). PERBANDINGAN PHONEGAP DAN REACT NATIVE SEBAGAI FRAMEWORK PENGEMBANGAN APLIKASI MOBILE. *Jurnal Manajemen Informatika & Sistem Informasi*, 1(2), 1-7.
- Yunandar, R. T., & Priyono. (2018). Pengujian Usability System Framework React Native dengan Expo untuk Pengembang Aplikasi Android Menggunakan Use Questionnaire. *Jurnal dan Penelitian Teknik Informatika - Sinkron*, 3(1), 252-259.