

Pengembangan Aplikasi Perangkat Bergerak: Pencari Konselor Psikologi Terdekat Berbasis Lokasi

Tifo Audi Alif Putra¹, Agi Putra Kharisma², Randy Cahya Wihandika³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹tifoaudi@student.ub.ac.id, ²agi@ub.ac.id, ³rendicahya@ub.ac.id

Abstrak

Kesehatan mental saat ini menjadi permasalahan yang tidak bisa diremehkan. Bahkan kesehatan mental sekarang menjadi masalah yang dihadapi banyak negara di dunia, tak terkecuali Indonesia. Depresi menjadi penyebab utama dalam permasalahan kesehatan mental. Depresi memengaruhi 4,4% populasi dunia dan 5% populasi di Indonesia. Selain jumlah penderita depresi yang tinggi, masalah lain pada Indonesia adalah kurangnya jumlah ketersediaan tenaga profesional kesehatan mental seperti konselor dan psikiater. Maka dari itu, pemanfaatan aplikasi berbasis *mobile* dapat dilakukan untuk menghubungkan orang yang memiliki permasalahan dengan konselor terdekat untuk mendapatkan pertolongan pertama dalam masalahnya. Implementasi pengembangan aplikasi perangkat bergerak pada penelitian ini dikembangkan pada platform *iOS* dengan menggunakan bahasa pemrograman *Swift*. Metode penelitian yang digunakan adalah *iterative waterfall* model yang terdiri dari analisis kebutuhan, perancangan sistem, implementasi sistem, dan pengujian. Pengujian pada penelitian ini terdiri dari *blackbox testing* dan *usability testing*. Pada *blackbox testing* peneliti merancang *testcase scenario* untuk menilai perilaku sistem yang benar sedangkan untuk mengukur tingkat *usability* dari aplikasi yang dikembangkan menggunakan instrumen SUPR-Qm. Hasil pengujian pada *blackbox testing* menyatakan 100% valid dari semua *testcase* yang dirancang. Untuk hasil pengujian *usability* diperoleh nilai 69,25% untuk aplikasi pasien dan 74,75% untuk aplikasi konselor, maka jika dikonversikan kedalam *usability rating scale* menghasilkan kategori *ok* untuk aplikasi pasien dan *good* untuk aplikasi konselor.

Kata kunci: kesehatan mental, depresi, *iOS*, *Swift*, *waterfall*, aplikasi perangkat bergerak.

Abstract

Mental health is now a problem that cannot be underestimated. Even mental health is now a problem faced by many countries in the world, including Indonesia. Depression is a major cause of mental health problems. Depression affects 4.4% of the world's population and 5% of the population in Indonesia. In addition to the high number of people with depression, another problem in Indonesia is the lack of available mental health professionals such as counselors and psychiatrists. Therefore, the use of mobile-based systems can be done to connect people who have problems with the nearest counselor to get first aid in the problem. The implementation of mobile application development in this study was developed on the iOS platform using the Swift programming language. The research method used is an iterative waterfall model consisting of needs analysis, system design, system implementation, and testing. Testing in this study consisted of blackbox testing and usability testing. In the blackbox testing the researchers designed a testcase scenario to assess the correct system behavior while to measure the usability level of the application developed using the SUPR-Qm instrument. The test results on Blackbox Testing stated 100% valid of all designed testcases. For the usability test results obtained 69.25% for patient applications and 74.75% for counselor applications, then if converted to usability rating scale produces an ok category for patient applications and good for counselor applications.

Keywords: *mental health, depression, iOS, Swift, waterfall, mobile-based system..*

1. PENDAHULUAN

Kesehatan mental saat ini menjadi permasalahan yang tidak bisa diremehkan. Bahkan kesehatan mental sekarang menjadi masalah yang dihadapi banyak negara di dunia, tak terkecuali Indonesia. Data dari WHO mengatakan bahwa 804.000 jumlah kematian pada tahun 2012 disebabkan oleh kasus bunuh diri, dimana rasio bunuh diri yang terjadi di Indonesia mencapai 4,3 per 100.000 penduduk (WHO, 2012). Penderita kesehatan mental paling banyak dialami oleh kalangan remaja atau mahasiswa dan memiliki dampak negatif yang serius terhadap prestasi belajar, keluarga, dan lingkungan sekitar (Chen & Jiang, 2019).

Depresi menjadi penyebab utama dalam permasalahan kesehatan mental. Depresi memengaruhi 4,4% populasi dunia dan 5% populasi di Indonesia. Selain jumlah penderita depresi yang tinggi, masalah lain pada Indonesia adalah kurangnya jumlah ketersediaan tenaga profesional kesehatan mental seperti konselor dan psikiater. WHO pun menginisiasi sebuah program bernama *Mental Health Gap Action Program* (mhGAP) yang bertujuan untuk menyediakan *internet-based intervention* untuk kesehatan mental yang dapat didistribusikan secara luas. (Arjadi, Nauta, & Bockting, 2018).

Berbagai upaya telah dilakukan oleh pemerintah Indonesia dalam menangani permasalahan kesehatan mental, salah satunya adalah menerbitkan UU no 18, tentang kesehatan mental dan perawatan orang dengan penyakit mental yang dicakup oleh cakupan kesehatan secara universal. Namun, banyak orang masih mengalami kesulitan dalam mengakses layanan kesehatan mental (Tristiana et al., 2018). Di sisi lain, terdapat penelitian yang menunjukkan bahwa orang yang merasa depresi, sedih atau kesepian cenderung melampiaskan permasalahan tersebut dengan menggunakan media sosial, salah satunya adalah *Twitter* untuk sekedar bercerita atau mendapat dukungan interaksi (Mahoney et al., 2019). Hal ini menunjukkan bahwa internet dan teknologi informasi mampu menghasilkan sebuah solusi yang menjanjikan untuk melakukan perawatan masalah kesehatan mental yang ada di Indonesia (Arjadi et al., 2016). Mayoritas orang Indonesia menunjukkan keterbukaan untuk menggunakan *internet-based intervention* dalam menangani permasalahan kesehatan mental, dimana sangat penting untuk dilakukan sosialisasi pemanfaatan internet kepada masyarakat (Arjadi et al., 2018).

Dengan perkembangan teknologi yang semakin canggih dan digital, mampu menciptakan peluang untuk menyelesaikan permasalahan tersebut menggunakan sistem berbasis *mobile*. Dari data yang diperoleh, jumlah masyarakat di Indonesia yang menggunakan *smartphone* mencapai 41 juta pengguna dengan sistem operasi Android, sedangkan untuk sistem operasi iOS mencapai 2,8 juta (Rahman, 2015). Dengan memanfaatkan fitur lokasi pengguna yang ada pada *smartphone*, maka memberi kemudahan pada pengguna untuk mengetahui lokasi pengguna lain dan dapat saling bertukar informasi. Maka dari itu, pemanfaatan aplikasi berbasis *mobile* dapat dilakukan untuk menghubungkan orang yang memiliki permasalahan dengan konselor terdekat untuk mendapatkan pertolongan pertama dalam masalahnya.

2. LANDASAN KEPUSTAKAAN

2.1. Penelitian Terkait

Dalam penelitian ini, penulis menemukan penelitian-penelitian serupa dan relevan yang sudah pernah dilakukan sehingga menjadi acuan dan referensi untuk penulis dalam melaksanakan penelitian.

Aplikasi pencarian rute perguruan tinggi berbasis android menggunakan *Location-Based Service*. Pada penelitian ini, penulis menawarkan solusi untuk memudahkan pengguna dalam mendapatkan informasi dan menentukan rute karena data dan informasi perguruan tinggi telah didaftarkan oleh peneliti (Nugroho, Nurhayati, & Widianto, 2017).

Aplikasi *mobile* berbasis lokasi untuk mencari layanan penyedia kesehatan di Yogyakarta. Pada penelitian ini, penulis membuat sebuah sistem yang mampu mencari lokasi layanan kesehatan dengan menggunakan *Google Map API* untuk menentukan rute dan lokasi terdekat (Erna Kumalasari Nurnawati, 2014).

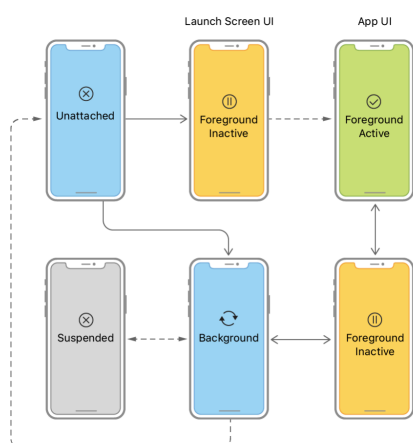
Pada negara Indonesia, *internet-based intervention* memiliki prospek yang menjanjikan dalam mengurangi kesenjangan kesehatan mental mengingat penggunaan internet dan *smartphone* terus meningkat tiap harinya. Untuk mengetahui apakah *internet-based intervention* diterima dan digunakan banyak orang, (Arjadi et al., 2018) melakukan investigasi untuk mengetahui faktor apa saja yang memengaruhi penggunaan *internet-based intervention* untuk

kesehatan mental di Indonesia. Hasil dari penelitian ini adalah mayoritas masyarakat Indonesia memiliki keterbukaan dalam menggunakan *internet-based intervention* dalam menangani permasalahan kesehatan mental di Indonesia. Untuk meningkatkan adopsi intervensi berbasis internet, penting untuk terlebih dahulu mempromosikan penggunaan internet kepada lebih banyak orang di seluruh negeri, terutama bagi mereka yang saat ini sedang mengalami gangguan kesehatan mental.

2.2. Mobile Application

Mobile application adalah perangkat lunak atau sekumpulan program yang berjalan pada *mobile device* dan menjalankan tugas tertentu untuk penggunaannya. *Mobile application* memiliki keunggulan diantaranya adalah kemudahan penggunaan, *user-friendly*, murah, mudah diunduh, dan berjalan hampir pada semua level perangkat bergerak. Selain itu, *mobile application* dapat digunakan untuk kebutuhan yang luas seperti menelpon, mengirim pesan, suara, *video*, *game*, dan lain-lain (Islam & Mazumder, 2010). Namun disamping kelebihan yang dimiliki, *mobile application* memiliki beberapa kelemahan diantaranya memiliki tampilan yang relatif kecil, memori yang terbatas, kemampuan komputasi pada CPU yang terbatas, rentang transfer data yang terbatas, dan lain-lain. (Oinas-Kukkonen & Kurkela, 2003)

2.2.1. iOS



Gambar 2.1 *Lifecycle* Aplikasi iOS
Sumber: Apple (2018)

iOS merupakan sistem operasi para perangkat mobile yang dibuat oleh Apple. Pada penelitian ini, sistem yang akan dibangun menggunakan bahasa *Swift* dan dibangun pada

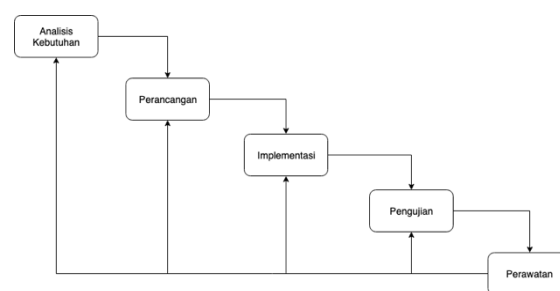
sistem operasi *iOS*. Diagram *lifecycle* aplikasi dengan sistem operasi *iOS* digambarkan pada Gambar 2.1

Adapun kelebihan pengembangan aplikasi pada sistem operasi *iOS* menurut (Avram, 2013) adalah sebagai berikut :

1. Apple sudah menyediakan standar dalam pengembangan *user-interface* sehingga pengembang perangkat lunak dapat menghemat waktu dalam melakukan pengembangan aplikasi.
2. Sistem operasi *iOS* hanya berjalan pada *iPhone* dan *iPad* sehingga pengembang dapat membuat aplikasi yang *responsive* dengan lebih mudah.
3. Keamanan data dan privasi pada aplikasi berbasis *iOS* dapat terjaga dengan baik.
4. Biaya pengembangan aplikasi berbasis *iOS* relatif lebih murah dan mudah.

2.3. Iterative Waterfall Model

Waterfall merupakan siklus pengembangan perangkat lunak yang bersifat linear. SDLC (*Software Development Lifecycle*) *Waterfall* dibagi menjadi dua diantaranya *Classic Waterfall* dan *Iterative Waterfall*. Kelemahan dari *Classic Waterfall* adalah sulit untuk mengatur perubahan dan perawatan pada sistem. *Iterative Waterfall* merupakan pengembangan lanjut dari *Classic Waterfall* untuk memudahkan tim dan *developer* agar mampu menghasilkan sistem dengan waktu yang efisien dan memudahkan dalam merawat sistem (Rather & Bhatnagar, 2016). Fase pengembangan pada *Iterative Waterfall* digambarkan pada Gambar 2.2.



Gambar 2.2 SDLC Waterfall
Sumber: Rather & Bhatnagar (2016)

2.4. Location-Based Service

Location-based services atau layanan berbasis lokasi adalah layanan yang memperhitungkan lokasi geografis dari suatu objek (Junglas & Watson, 2008). Dengan adanya fitur *Global Positioning System* (GPS) pada

perangkat *mobile* dapat membantu pengguna dalam melihat informasi lokasi suatu objek. Dalam konteks penelitian ini, peneliti menggunakan layanan berbasis lokasi dalam aplikasi untuk menentukan dan mencari konselor psikologi terdekat dengan pengguna.

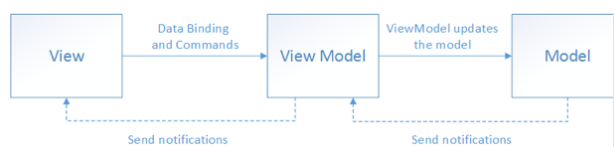
2.5. Best Practice

2.5.1. Swift

Swift adalah bahasa pemrograman baru yang dibuat oleh Apple pada tahun 2010 dan diumumkan pada tahun 2015 yang tujuannya utamanya digunakan untuk mengembangkan program di atas platform Apple (iOS, macOS, tvOS dan watchOS). Swift menawarkan kelebihan baru seperti *Type Safety*, *Fast* dan *Expressive*. *Type safety* yang adalah cara pemrograman baru agar program berjalan dengan cara yang aman (menghindari *null pointer*). Swift digunakan untuk menggantikan bahasa pemrograman yang dipakai Apple sebelumnya, yaitu Objective-C, yang secara performa lebih cepat dari bahasa pendahulunya. (Apple, n.d.-b)

2.5.2. MVVM (Model-View-ViewModel)

MVVM merupakan arsitektur perangkat lunak yang memiliki tiga layer yaitu model, view, dan view model. Dengan adanya tiga layer dalam sistem maka terdapat pemisahan antara *business-logic*, *presentation-logic*, dan *UI-logic*. Arsitektur ini awalnya dikembangkan oleh Microsoft untuk mengembangkan aplikasi pada platform mereka, kemudian banyak diadopsi oleh pengembang pada platform lainnya. Adapun alur interaksi antar *layer* digambarkan pada Gambar 2.3.



Gambar 2.1 Arsitektur MVVM
Sumber: Microsoft (2017)

2.5.3. MapKit

MapKit merupakan framework yang dibuat oleh Apple untuk menampilkan peta. Pada versi iOS 5.1 atau yang terbaru, MapKit menggunakan *Google Mobile Maps* (GMM) untuk menyediakan data yang sudah terintegrasi (Apple, n.d.-a). Pada penelitian ini, penulis menggunakan MapKit untuk menyediakan

informasi lokasi konselor psikologi pada pengguna untuk memudahkan interaksi keduanya.

2.5.4. Firebase

Firebase merupakan penyedia layanan berbasis *cloud* yang dibuat oleh Google. Dengan adanya Firebase, pengembang perangkat lunak tidak perlu memikirkan untuk membuat *web-service* atau API untuk melakukan pengolahan data. Layanan Firebase yang digunakan pada penelitian ini adalah *Realtime Database* dan *Firebase Authentication*.

Implementasi Firebase pada pengembangan aplikasi perangkat bergerak memiliki keuntungan. Menurut (Mevada, 2018) keuntungan menggunakan firebase adalah sebagai berikut:

1. Penyimpanan data secara *real-time* dan sinkronisasi data
2. Adanya *Google Analytics* yang dapat membantu pengembang untuk melacak perilaku dari pengguna.
3. Firebase menyediakan fitur *crash reporting* pada pengguna sehingga pengembang dapat memperbaiki bug tersebut dengan cepat.
4. Firebase Authentication yang mempermudah pengembang dalam membuat fitur autentikasi pada aplikasi.

2.6 Pengujian Perangkat Lunak

2.6.1 Blackbox Testing

BlackBox Testing adalah pengujian yang dilakukan untuk mengevaluasi fitur atau kebutuhan dari pengguna dan memastikan bahwa sistem berjalan sesuai semestinya (Larrea, 2017). *Test-Case* yang dirancang pada blackbox testing bergantung pada spesifikasi perangkat lunak yang telah ditetapkan.

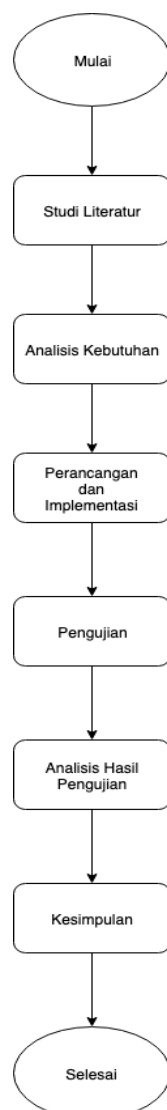
2.6.2 Usability Testing

Usability Testing adalah pengujian yang dilakukan untuk menilai kualitas dari sistem oleh calon pengguna utama. Tujuan dilakukan usability testing adalah menemukan permasalahan yang dialami calon pengguna agar bisa diperbaiki untuk meningkatkan kenyamanan pengguna dalam mengoperasikan sistem nantinya. Tingkat usability dapat diketahui dengan melakukan pengukuran tentang sejauh mana tingkat kemudahan dan tingkat kepuasan dari pengguna (Farouqi, Aknuranda, & Herlambang, 2018).

Jumlah responden untuk usability testing dipilih 5 orang. Alasan pemilihan jumlah responden tersebut adalah 5 orang responden merupakan angka yang optimal dan efektif dimana mengacu pada keuntungan dan usaha yang dilakukan dalam pengujian (Nielson & Landauer, 1993). Alasan lain dipilih 5 orang sebagai jumlah responden adalah 95% permasalahan yang ada pada sistem akan muncul pada 5 orang pertama ketika dilakukan usability testing (Cao, 2015).

3. METODOLOGI

Dalam sebuah penelitian diperlukan metodologi untuk menjadi landasan dan pedoman agar penelitian yang dilakukan terstruktur rapi. Metode yang digunakan pada penelitian yang dilakukan adalah *Waterfall*. Alur penelitian yang akan dilakukan digambarkan pada Gambar 3.1



Gambar 3.1 Diagram Alir Penelitian

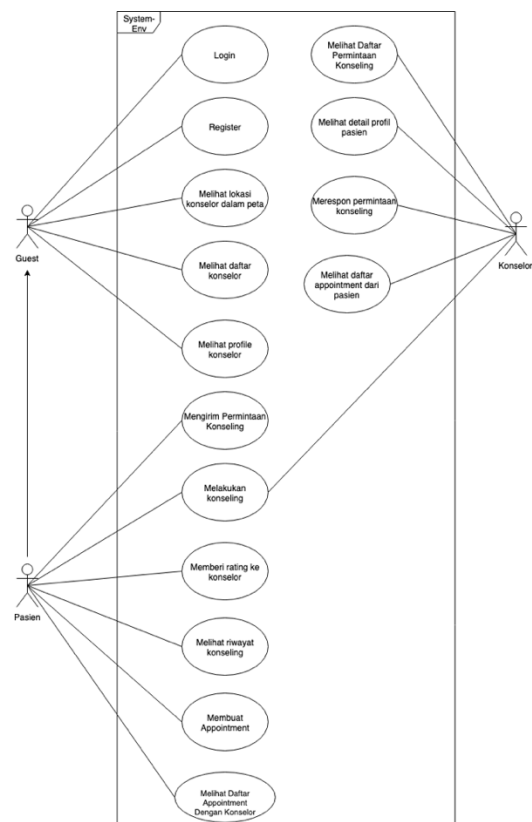
4. ANALISIS KEBUTUHAN

Pada bab ini memuat proses analisis kebutuhan terhadap calon pengguna aplikasi yang berguna untuk mendapatkan kebutuhan sistem sehingga proses pengembangan dapat dimulai. Kebutuhan sistem yang dibuat mengacu pada hasil wawancara yang dilakukan terhadap calon pengguna dan konselor.

Berdasarkan hasil analisis kebutuhan sistem dilakukan identifikasi aktor yang dapat dilihat pada Tabel 4.1.

Tabel 4.1 Identifikasi Aktor

Aktor	Deskripsi
Guest	Pada dasarnya, guest adalah pengguna namun belum memiliki akun atau belum melakukan autentikasi. Aktor guest dapat melihat aplikasi namun memerlukan proses autentikasi untuk mengakses fitur-fitur tertentu.
Pasien	Pasien adalah pengguna yang telah teridentifikasi melalui proses autentikasi pada aplikasi. Aktor pasien memiliki wewenang untuk mengakses semua fitur dari aplikasi yang dikembangkan.
Konselor	Aktor konselor adalah aktor yang telah teridentifikasi sebagai konselor setelah melakukan proses autentikasi.



Gambar 4.1 Use Case Diagram

Usecase diagram adalah diagram yang menggambarkan interaksi secara abstrak antara aktor dengan sistem (Shivaram & Handigund, 2015). Pada penelitian ini terdapat tiga aktor dengan masing-masing peran yang berbeda. Usecase diagram pada penelitian ini dapat dilihat pada Gambar 4.1. Pada usecase diagram yang telah dibuat, terdapat tiga aktor yang terlibat didalam sistem yaitu guest, pasien, dan konselor.

5. PERANCANGAN

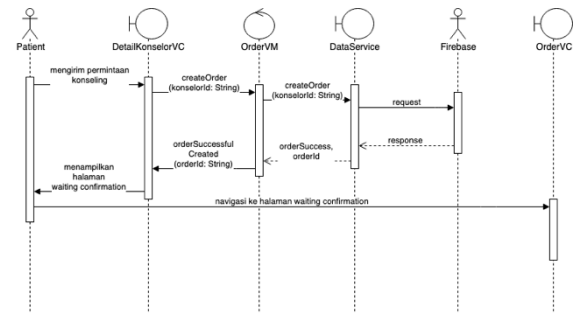
Bab ini membahas tentang perancangan yang dilakukan untuk membuat gambaran dari sistem yang akan dibuat. Hasil dari perancangan akan diimplementasikan dan dibahas pada bab implementasi.

Pada arsitektur sistem yang dibangun terdapat tiga komponen *library* utama yaitu *MapKit*, *CLLocation*, dan *Firestore*. *MapKit* adalah framework buatan Apple yang digunakan untuk menampilkan peta. Dalam hal ini, *MapKit* digunakan untuk menampilkan lokasi pasien dan konselor psikologi terdekat dengan pasien. *CLLocation* adalah *native-library* yang digunakan untuk mengatur dan memonitor lokasi terkini dari pasien dan mendapatkan alamat fisik dari konselor psikologi. *Firestore* adalah layanan berbasis *cloud* yang dibuat oleh Google. Pada penelitian ini, layanan *Firestore* yang digunakan adalah *Realtime Database* yang bertujuan untuk melakukan transaksi pertukaran data dengan sisi aplikasi serta memanfaatkan fitur *realtime* untuk membuat layanan *chatting* pada sisi aplikasi.

Selain menggunakan ketiga *library* tersebut, arsitektur yang digunakan untuk mengorganisir tiap modul dan fitur adalah *MVVM (Model-View-ViewModel)*. *Model* memiliki tugas untuk membuat representasi data objek. *ViewModel* memiliki tugas untuk menyediakan data yang siap dikonsumsi untuk *view*. Dan *view* memiliki tugas untuk menampilkan data dan menerima respon atau *event* dari pengguna. Dalam penerapannya, terdapat penambahan satu layer yaitu *Service*, dimana sebuah objek *singleton* yang memiliki tugas untuk melakukan *request* ke *Firestore* untuk melakukan pertukaran data.

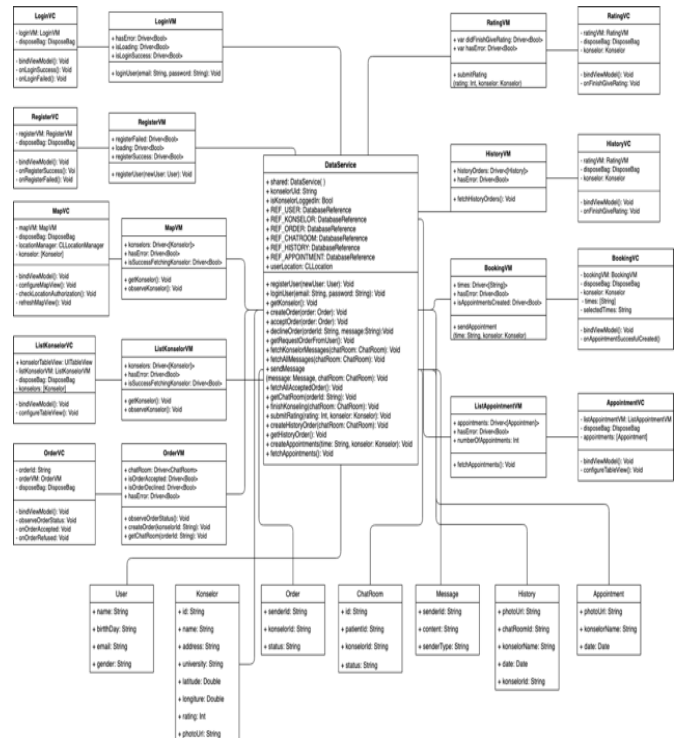
Sequence diagram merupakan diagram yang menggambarkan urutan aktifitas antara aktor dengan komponen pada sistem pada masing-masing kebutuhan fungsional. Berikut adalah contoh hasil perancangan *sequence diagram* pada salah satu kebutuhan fungsional pada aplikasi yang dikembangkan digambarkan

pada Gambar 5.1

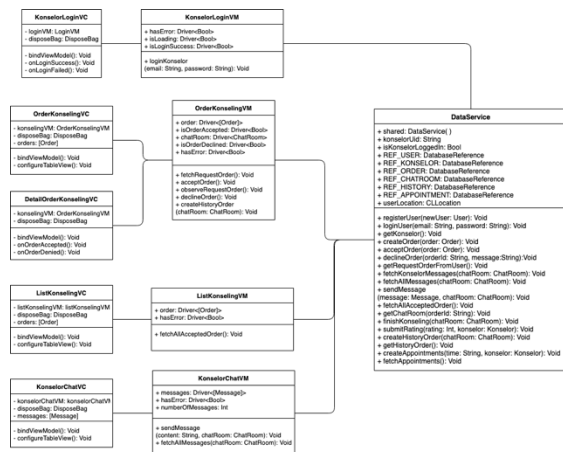


Gambar 5.1 *Sequence Diagram* Mengirim Permintaan Konseling

Class Diagram adalah diagram yang menggambarkan arsitektur objek dan *class* pada sistem beserta hubungan keterkaitan antar satu dengan yang lain dan menjelaskan bagaimana tiap modul dikelompokkan. *Class Diagram* yang dibuat penelitian ini dikelompokkan menjadi 2, yaitu *Class Diagram* untuk aplikasi sisi pasien dan *Class Diagram* untuk aplikasi sisi konselor. *Class Diagram* untuk pasien digambarkan pada Gambar 5.2 dan *Class Diagram* untuk konselor digambarkan pada Gambar 5.3



Gambar 5.2 *Class Diagram* Pasien



Gambar 5.3 Class Diagram Konselor

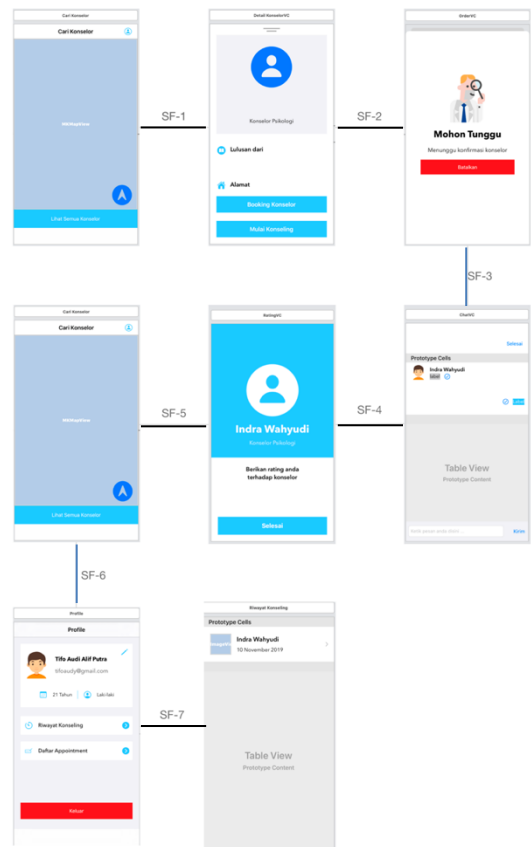
Untuk Perancangan tabel data untuk entitas Konselor dapat dilihat pada Tabel 5.1.

Tabel 5.1 Rancangan Data Konselor

No	Nama Field	Deskripsi
1.	name	Merupakan nama lengkap pasien
2.	email	Merupakan alamat email dari pasien
3.	gender	Merupakan jenis kelamin dari pasien
4.	birthday	Merupakan tanggal lahir dari pasien
5.	latitude	Merupakan satuan posisi latitude dari pasien
6.	longitude	Merupakan satuan posisi longitude dari pasien

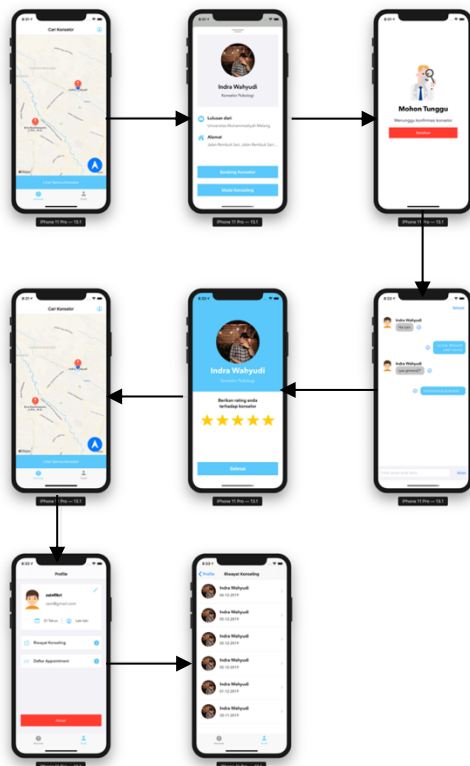
Perancangan algoritme dilakukan untuk membuat prosedur yang berisi langkah-langkah yang diperlukan dalam menyelesaikan sebuah permasalahan. Dalam aplikasi yang dikembangkan pada penelitian ini, terdapat 3 algoritme fitur utama pada aplikasi

Kemudian dilakukan perancangan antarmuka berupa *screen-flow* agar memudahkan dalam memahami alur sistem dalam membangun user interface. Adapun perancangan antarmuka pada aplikasi sisi pasien digambarkan pada Gambar 5.4



Gambar 5.4 Screenflow Pasien

6. IMPLEMENTASI



Gambar 6.1 Implementasi Aplikasi

Pada tahap ini telah dilakukan implementasi untuk membangun aplikasi dari hasil perancangan yang telah didefinisikan pada tahap sebelumnya. Implementasi dilakukan mulai dari implementasi perancangan basis data, implementasi kode program, dan implementasi antarmuka. Adapun hasil implementasi aplikasi yang dikembangkan digambarkan pada Gambar 6.1

7. PENGUJIAN

Pada bab pengujian menjelaskan tentang *blackbox testing* dan pengujian usabilitas pada aplikasi. *Blackbox testing* dilakukan di setiap fitur untuk memastikan aplikasi berjalan dengan sesuai dan memastikan tidak ada *bug*. *Blackbox testing* dilakukan dengan cara menyediakan *input* dan memastikan *output* sesuai dengan harapan. Pengujian usabilitas dilakukan untuk mengukur tingkat kemudahan dalam penggunaan aplikasi.

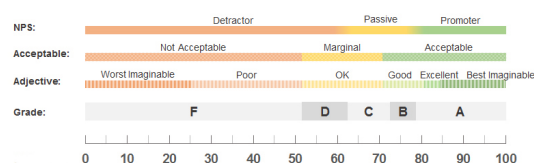
Pada pengembangan aplikasi yang dilakukan terdapat 4 fitur yang akan dilakukan pengujian *blackbox*. Pengujian *blackbox* yang dilakukan dimulai dari peneliti merancang *testcase* dan menguji aplikasi sesuai *testcase* tersebut untuk mengetahui apakah perilaku sistem valid atau tidak. Hasil yang diperoleh adalah 100% valid untuk semua *testcase* dari 4 fitur yang dilakukan pengujian.

Pengujian usabilitas merupakan pengujian yang dilakukan untuk mengukur tingkat kemudahan dalam penggunaan sistem. Jumlah responden yang ditentukan pada penelitian ini adalah 5 orang untuk aplikasi sisi pasien dan 5 orang untuk aplikasi sisi konselor. Langkah pengujian usabilitas dimulai ketika responden melakukan semua skenario pengujian yang berisi tugas-tugas yang berbeda dalam mengoperasikan aplikasi. Setelah melakukan semua skenario pengujian, responden diminta untuk mengisi kuisioner untuk memberikan penilaian terhadap aplikasi yang dikembangkan. Instrumen pertanyaan yang digunakan pada pengisian kuisioner menggunakan SUPR-Qm dimana tiap pertanyaan menggunakan skala likert sebagai skala penilaian.

Setelah selesai dalam mengisi kuisioner, responden dapat memberikan kritik dan saran terkait aplikasi yang dikembangkan. Kritik dan saran tersebut sangat dibutuhkan untuk menjadi umpan balik pengembang dalam mengembangkan aplikasi lebih baik lagi sehingga dapat dilanjutkan implementasinya

pada penelitian lain. Pengujian usabilitas dilakukan oleh 5 responden pada masing-masing aktor yaitu aplikasi pada sisi pasien dan aplikasi pada sisi konselor. Responden melakukan skenario yang telah ditentukan dari masing-masing aplikasi untuk mengukur kemudahan dari aplikasi yang dikembangkan.

Dari hasil pengujian *usability* yang dilakukan, diperoleh nilai 69,25% untuk aplikasi pada sisi pasien dan 74,5% untuk aplikasi pada sisi konselor. Setelah mengisi pertanyaan yang ada pada SUPR-Qm, responden memberikan kritik dan saran pada aplikasi yang dikembangkan agar dilakukan pengembangan lebih baik lagi. Setelah dilakukan pengujian *usability*, maka dapat diperoleh nilai atau skor SUPR-Qm. Agar dapat menyimpulkan hasil berdasarkan skor tersebut, maka perlu dilakukan konversi skor tersebut ke *usability rating scale*.



Gambar 7.1 Usability Rating Scale

Skor SUPER-Qm dari aplikasi sisi pasien adalah 69,25% maka dapat disimpulkan hasil rating pada aplikasi sisi pasien adalah masuk dalam kategori *ok*. Untuk skor SUPER-Qm dari aplikasi sisi konselor adalah 74,5% maka hasil rating dari aplikasi sisi konselor juga masuk dalam kategori *good*.

8. KESIMPULAN DAN SARAN

Setelah penelitian yang dilakukan yang dimulai dari tahap analisis kebutuhan, perancangan, implementasi, dan pengujian didapatkan hasil kesimpulan sebagai berikut:

1. Tahap analisis kebutuhan pada penelitian ini menghasilkan 14 kebutuhan fungsional dimana terdiri dari 10 kebutuhan fungsional untuk aplikasi sisi pasien dan 4 kebutuhan fungsional untuk aplikasi sisi konselor. Selain kebutuhan fungsional, terdapat kebutuhan non-fungsional yang didefinisikan pada penelitian ini. Kebutuhan fungsional aplikasi sisi pasien pada penelitian ini terdiri dari proses login, register, menampilkan peta berisi lokasi pasien dan konselor, menampilkan daftar semua konselor, menampilkan profil konselor, mengirim permintaan konseling,

konseling secara online, memberi rating kepada konselor, menampilkan halaman riwayat konseling, dan membuat appointment dengan konselor. Kebutuhan fungsional pada aplikasi sisi konselor terdiri dari menampilkan seluruh permintaan konseling, fasilitas untuk menerima atau menolak permintaan konseling, menampilkan profil pasien, dan menampilkan daftar appointment dengan pasien. Sedangkan kebutuhan non-fungsional terdiri *usability* yang diukur dengan melalui skema pengujian usabilitas. Kebutuhan tersebut didapatkan dari proses wawancara personal dengan pengguna dan wawancara dengan konselor psikologi.

2. Tahap perancangan sistem pada penelitian ini menghasilkan *class diagram*, *sequence diagram*, rancangan algoritme, rancangan antarmuka sistem, dan rancangan tabel data yang menggunakan struktur data *tree*.
3. Tahap implementasi pada penelitian ini menghasilkan aplikasi perangkat bergerak pada sistem operasi iOS yang memiliki fitur sesuai dengan hasil analisis kebutuhan yang telah dilakukan dengan menggunakan bahasa pemrograman *Swift*.
4. Tahap pengujian pada penelitian ini terdiri dari 2 pengujian yaitu pengujian blackbox dan pengujian usability. Hasil dari pengujian blackbox adalah 100% valid dan hasil pengujian usability untuk aplikasi sisi pasien masuk ke dalam kategori *ok* dan aplikasi sisi konselor masuk dalam kategori *good*.

Saran yang didapat dari pengembangan aplikasi perangkat bergerak untuk mencari konselor psikologi terdekat adalah sebagai berikut:

1. Implementasi notifikasi untuk memudahkan proses konseling atau interaksi antara pasien dengan konselor.
2. Memberikan informasi yang lebih lengkap pada profil pasien agar memudahkan konselor.
3. Adanya fitur tambahan seperti *video call* atau telfon pada konseling *online*.
4. Perbaikan *user interface* dan *user experience* agar aplikasi lebih menarik dan lebih mudah digunakan.

9. DAFTAR PUSTAKA

- Apple. (n.d.-a). MapKit. Retrieved October 6, 2019, from <https://developer.apple.com/documentation/mapkit>
- Apple. (n.d.-b). Swift The powerful programming language that is also easy to learn. Retrieved October 5, 2019, from <https://developer.apple.com/swift/>
- Apple. (2018). Managing Your App's Life Cycle. Retrieved August 24, 2019, from https://developer.apple.com/documentation/uikit/app_and_environment/managing_your_app_s_life_cycle
- Arjadi, R., Nauta, M. H., & Bockting, C. L. H. (2018). Acceptability of internet-based interventions for depression in Indonesia. *Internet Interventions*, 13(April), 8–15. <https://doi.org/10.1016/j.invent.2018.04.004>
- Arjadi, R., Nauta, M. H., Scholte, W. F., Hollon, S. D., Chowdhary, N., Suryani, A. O., & Bockting, C. L. H. (2016). Guided Act and Feel Indonesia (GAF-ID) - Internet-based behavioral activation intervention for depression in Indonesia: Study protocol for a randomized controlled trial. *Trials*, 17(1), 1–10. <https://doi.org/10.1186/s13063-016-1577-9>
- Avram, A. (2013). iOS vs. Android Development. Retrieved October 5, 2019, from InfoQ website: <http://www.infoq.com/news/2013/08/ios-vs-android-development>
- Bajpai, V., & Gorthi, R. P. (2012). On non-functional requirements: A survey. 2012 IEEE Students' Conference on Electrical, Electronics and Computer Science: Innovation for Humanity, SCEECS 2012, (April). <https://doi.org/10.1109/SCEECS.2012.6184810>
- Cao, J. (2015). The Guide to Usability Testing.
- Chen, M., & Jiang, S. (2019). Analysis and research on mental health of college students based on cognitive computing. *Cognitive Systems Research*, 56, 151–158. <https://doi.org/10.1016/j.cogsys.2019.03.003>
- Erna Kumalasari Nurnawati, J. M. (2014). Aplikasi mobile berbasis lokasi untuk penyedia lokasi layanan kesehatan di

- Yogyakarta. Prosiding Seminar Nasional Aplikasi Sains & Teknologi (SNAST), 293–300.
- Faisandier, A. (2012). SYSTEMS OPPORTUNITIES AND REQUIREMENTS.
- Farouqi, M. I., Aknuranda, I., & Herlambang, A. D. (2018). Evaluasi Usability pada Aplikasi Go-Jek Dengan Menggunakan Metode Pengujian Usability. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 2(10), 3150–3156. Retrieved from <http://j-ptiik.ub.ac.id/index.php/j-ptiik/article/download/2396/947/>
- Hidalgo, E. S. (2019). Adapting the scrum framework for agile project management in science: case study of a distributed research initiative. *Heliyon*, 5(3), e01447. <https://doi.org/10.1016/j.heliyon.2019.e01447>
- Islam, R., & Mazumder, T. (2010). Mobile application and its global impact. *International Journal of Engineering & ...*, (06), 72–78. Retrieved from <http://ijens.org/107506-0909 IJET-IJENS.pdf>
- Junglas, B. I. A., & Watson, R. T. (2008). LOCATION-BASED SERVICES. 51(3), 65–70.
- Larrea, M. (2017). Black-Box Testing Technique for Information Visualization. Sequencing Constraints with Low-Level Interactions. *Journal of Computer Science and Technology*, 17(1), 37–48.
- Mahoney, J., Le Moignan, E., Long, K., Wilson, M., Barnett, J., Vines, J., & Lawson, S. (2019). Feeling alone among 317 million others: Disclosures of loneliness on Twitter. *Computers in Human Behavior*, 98(February), 20–30. <https://doi.org/10.1016/j.chb.2019.03.024>
- Mevada, D. (2018). The Benefits of Having Firebase for Mobile App Development. Retrieved September 6, 2019, from <https://www.mindinventory.com/blog/benefits-of-firebase-in-mobile-app-development/>
- Microsoft. (2017). The Model-View-ViewModel Pattern. Retrieved October 5, 2019, from <https://docs.microsoft.com/en-us/xamarin/xamarin-forms/enterprise-application-patterns/mvvm>
- Nielson, J., & Landauer, J. (1993). A mathematical model of finding the usability problem. *Proceedings of the CHI 93 proceedings of the Interact conference on human factors in computing systems. Espacio de Trabajo Matemático. Quinto Simposio Internacional ET*, 206–213. <https://doi.org/10.1145/169059.169166>
- Nugroho, S. C., Nurhayati, O. D., & Widiyanto, E. D. (2017). Aplikasi Pencarian Rute Perguruan Tinggi Berbasis Android Menggunakan Location Based Service (LBS) di Kota Semarang. *Jurnal Teknologi Dan Sistem Komputer*, 3(2), 311. <https://doi.org/10.14710/jtsiskom.3.2.2015.311-319>
- Oinas-Kukkonen, H., & Kurkela, V. (2003). Developing successful mobile applications. *Proceedings of the IASTED International Conference on Computer Science and Technology*, (January 2003), 50–54.
- Rahman. (2015). Android Kuasai Asia Tenggara, di Indonesia Paling Juara.
- Rather, M. A., & Bhatnagar, V. (2016). A comparative study of sdlc model. (October 2015).
- Sauro, J., & Zarolia, P. (2017). SUPR-Qm: A Questionnaire to Measure the Mobile App User Experience. *Journal of Usability Studies*, 13(1), 17–37.
- Shivaram, A. M., & Handigund, S. M. (2015). An ameliorated methodology for the abstraction of object oriented features from software requirements specification. *Procedia Computer Science*, 62(Scse), 274–281. <https://doi.org/10.1016/j.procs.2015.08.450>
- Tristiana, R. D., Yusuf, A., Fitryasari, R., Wahyuni, S. D., & Nihayati, H. E. (2018). Perceived barriers on mental health services by the family of patients with mental illness. *International Journal of Nursing Sciences*, 5(1), 63–67. <https://doi.org/10.1016/j.ijnss.2017.12.003>
- WHO. (2012). Suicide mortality rate (per 100,000 population).