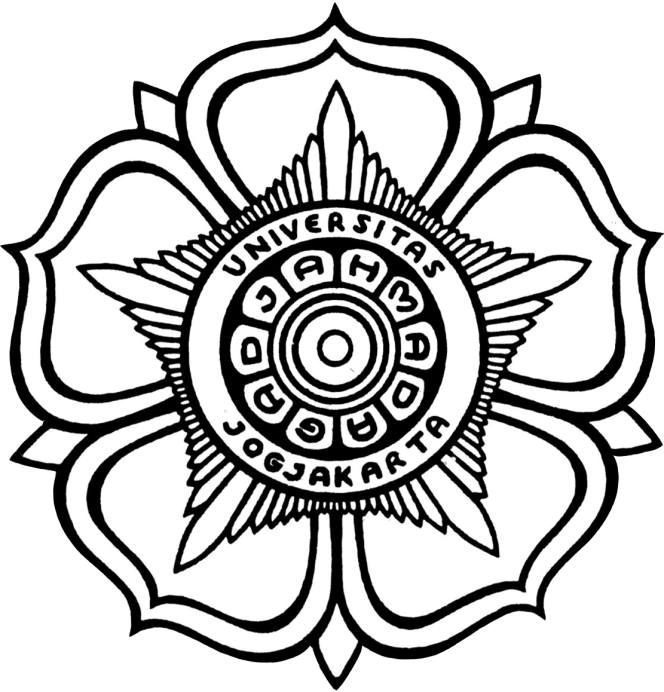
SKRIPSI



Disusun oleh :

MEIZAR RAKA RIMADANA

14/363478/TK/41594

MAXIMILLIAN SHELDY FERDINAND ERWIANDA

**14/365238/TK/42057**

PROGRAM STUDI TEKNOLOGI INFORMASI

DEPARTEMEN TEKNIK ELEKTRO DAN TEKNOLOGI INFORMASI

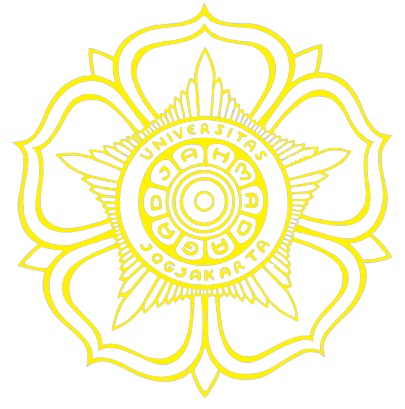
FAKULTAS TEKNIK UNIVERSITAS GADJAH MADA

YOGYAKARTA

HALAMAN PENGESAHAN

ANALISIS UJI KORELASI ANTARA DATA PENGGUNAAN APLIKASI SMARTPHONE TERHADAP MODEL GAYA BELAJAR VARK (VISUAL, AURAL, READ/WRITE, KINESTHETIC)

SKRIPSI



Diajukan Sebagai Salah Satu Syarat untuk Memperoleh

Gelar Sarjana Teknik Program S-1

Pada Departemen Teknik Elektro dan Teknologi Informasi Fakultas Teknik

Universitas Gadjah Mada

Disusun oleh :

MEIZAR RAKA RIMADANA

14/363478/TK/41594

MAXIMILLIAN SHELDY FERDINAND ERWIANDA

14/365238/TK/42057

Telah disetujui dan disahkan

pada tanggal <tanggal lulus ujian pendadaran>

Dosen Pembimbing I

**P. Insap Santosa, Ir., M.Sc., Ph.D.**

**NIP. 196101081985031002**

Dosen Pembimbing II

**Sri Suning Kusumawardani, Dr., S.T., M.T.**

**NIP. 196911221995122001**

LEMBAR PERNYATAAN

Dengan ini, saya menyatakan bahwa dalam skripsi ini tidak terdapat karya dengan judul serupa yang pernah diajukan untuk memperoleh gelar sarjana di suatu perguruan tinggi dan saya menjamin keaslian dari hasil karya skripsi ini. Jika terdapat karya orang lain yang dimuat dalam skripsi ini, saya akan mencantumkan sumber yang jelas. Dan apabila kelak di kemudian hari ditemukan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima konsekuensi sesuai dengan aturan yang berlaku. Demikian pernyataan ini saya buat dengan sebenarnya tanpa ada paksaan dari pihak manapun.

Yogyakarta, 26 Juni 2018

Meizar Raka Rimadana

LEMBAR PERNYATAAN

Dengan ini, saya menyatakan bahwa dalam skripsi ini tidak terdapat karya dengan judul serupa yang pernah diajukan untuk memperoleh gelar sarjana di suatu perguruan tinggi dan saya menjamin keaslian dari hasil karya skripsi ini. Jika terdapat karya orang lain yang dimuat dalam skripsi ini, saya akan mencantumkan sumber yang jelas. Dan apabila kelak di kemudian hari ditemukan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima konsekuensi sesuai dengan aturan yang berlaku. Demikian pernyataan ini saya buat dengan sebenarnya tanpa ada paksaan dari pihak manapun.

Yogyakarta, 26 Juni 2018

Maximillian Sheldy Ferdinand Erwianda

HALAMAN PERSEMBAHAN

Ku persembahkan Skripsi ini untuk yang selalu bertanya:

“kapan Skripsimu selesai?”

Terlambat lulus atau lulus tepat waktu bukan sebuah kejahatan,

Bukan sebuah aib. Alangkah kerdilnya jika mengukur kepintaran seseorang hanya dari siapa yang paling cepat lulus. Bukankah sebaik-baiknya Skripsi adalah Skripsi yang selesai? Baik itu selesai tepat waktu maupun tidak tepat waktu.

*- Anonymous*, 2018 -

KATA PENGANTAR

Puji syukur kehadirat Allah SWT atas limpahan rahmat, karunia, serta hidayahNya sehingga tugas akhir berupa penelitian dan penyusunan skripsi ini telah dapat diselesaikan dengan baik.

Dalam penyusunan tugas akhir ini penulis telah banyak mendapatkan arahan, bantuan, serta dukungan dari berbagai pihak. Oleh karena itu pada kesempatan ini penulis mengucapkan terima kasih kepada:

1. Bapak Sarjiya, S.T., M.T., Ph.D, selaku Ketua Departemen Teknik Elektro dan Teknologi Informasi Fakultas Teknik Universitas Gadjah Mada.
2. Bapak Ridi Ferdiana, Dr., S.T., M.T., selaku dosen pembimbing pertama yang telah memberikan ilmu, arahan, bimbingan, saran, dan pengertian yang diberikan kepada penulis dalam menyelesaikan tugas akhir ini.
3. Bapak Rudy Hartanto, Dr. Ir., M.T., selaku dosen pembimbing kedua yang telah memberikan ilmu, arahan, bimbingan, saran, dan pengertian yang diberikan kepada penulis dalam menyelesaikan tugas akhir ini.
4. Ibu Sri Suning Kusumawardani, Dr., S.T., M.T., selaku dosen pembimbing akademik atas nasihat dan bimbingannya selama masa perkuliahan.
5. Bapak Selo, S.T., M.T., M.Sc., Ph.D., selaku dosen pembimbing kerja praktik atas nasihat dan bimbingannya selama masa kerja praktik.
6. Bapak Bimo Sunarfri Hantono, S.T., M.Eng., selaku penanggung jawab kerja praktik atas kepercayaan, nasihat, dan bimbingan selama pengerjaan kerja praktik.
7. Seluruh Bapak dan Ibu Dosen serta Karyawan DTETI FT UGM yang telah memberikan lingkungan yang sangat ideal untuk menimba ilmu dan berkarya.
8. Kedua orang tua tercinta, Bapak Muhammad Zainuri dan Ibu Tuti Setyawati, yang senantiasa selalu mendukung untuk maju, sehingga penulis berani untuk bermimpi besar dan selalu berusaha untuk menggapainya. Tanpa doa dan restunya penulis tidak akan dapat menggapai berbagai hal dalam perkuliahan ini.
9. Kepada kakak penulis, Himma Illiyana yang selalu memberikan motivasi, arahan, dan semangat. Terima kasih atas segala dorongan dan tuntunan yang amat luar biasa bermanfaat bagi kehidupan penulis.
10. Kepada Yona dan Bertus, terima kasih telah menjadi rekan kerja praktik ARme yang sangat berkesan.
11. Kepada Ridho, Niko, Gingsir, Yonif, dan Mas Ubai, terima kasih telah membantu dalam proses pengujian aplikasi dan penulisan skripsi. Tanpa bantuan kalian, penulis tidak dapat menyelesaikan penelitian tepat waktu.
12. Kepada Iqbal, Fathin, Fauzi, Nadya, Niko, dan Zaza, terima kasih telah memberikan pengalaman kepada penulis untuk berkompetisi. Semoga ilmu yang didapatkan dapat berguna.
13. Kepada teman-teman GMRR 21 yang selalu membuat tawa dengan pemikiran yang tidak biasa dan senantiasa ada dikala suka maupun duka, semoga makin akur.
14. Kepada kolega di Lab. Informatika serta laboran, terima kasih untuk waktu-waktunya. Di lab ini saya bersapa dan bertemu dengan teman-teman yang giat menimba ilmu pun telah menghidupkan suasana lab.
15. Kepada teman-teman The A Team, AIESEC LC UGM, terima kasih untuk 1 tahun periode yang diberikan sehingga penulis dapat belajar untuk berorganisasi dan dapat bertemu dengan orang-orang hebat yang inspiratif.
16. Kepada teman-teman KKN-PPM JTG 72 Desa Pegundan, Pemalang, terima kasih atas pengalaman bersosialisasi dengan masyarakat. Semoga ilmu yang kita dapatkan dapat bermanfaat.
17. Para ilmuwan dan peneliti di bidang *Augmented Reality*.

Yogyakarta, 2 April 2018

Penulis

DAFTAR ISI

[HALAMAN JUDUL i](#_Toc517886255)

[HALAMAN PENGESAHAN ii](#_Toc517886256)

[LEMBAR PERNYATAAN iii](#_Toc517886257)

[HALAMAN PERSEMBAHAN v](#_Toc517886260)

[KATA PENGANTAR vi](#_Toc517886261)

[DAFTAR ISI ix](#_Toc517886262)

[DAFTAR TABEL xii](#_Toc517886263)

[DAFTAR GAMBAR xiii](#_Toc517886264)

[DAFTAR SINGKATAN xix](#_Toc517886265)

[INTISARI xxi](#_Toc517886266)

[*ABSTRACT* xxii](#_Toc517886267)

[BAB I PENDAHULUAN 1](#_Toc517886268)

[1.1. *Latar Belakang* 1](#_Toc517886269)

[1.2. *Rumusan Masalah* 4](#_Toc517886270)

[1.3. *Keaslian Penelitian* 4](#_Toc517886271)

[1.4. *Batasan Masalah* 7](#_Toc517886272)

[1.5. *Tujuan Penelitian* 7](#_Toc517886273)

[1.6. *Manfaat Penelitian* 8](#_Toc517886274)

[1.7. *Sistematika Penulisan* 8](#_Toc517886275)

[BAB II TINJAUAN PUSTAKA DAN DASAR TEORI 10](#_Toc517886276)

[2.1. *Tinjauan Pustaka* 10](#_Toc517886277)

[2.2. *Dasar Teori* 13](#_Toc517886278)

[2.2.1. Belajar 13](#_Toc517886279)

[2.2.2. Model Gaya Belajar VARK 15](#_Toc517886280)

[2.2.3. *Smartphone* 16](#_Toc517886281)

[2.2.4. Aplikasi Peranti Bergerak 18](#_Toc517886282)

[2.2.5. Android 19](#_Toc517886283)

[2.2.6. App Usage Statistic API 21](#_Toc517886284)

[2.2.7. Firebase 22](#_Toc517886285)

[2.2.8. Evernote Android Job 24](#_Toc517886286)

[2.2.9. Metode Pengembangan Perangkat Lunak SCRUM 24](#_Toc517886287)

[2.2.10. Uji Korelasi Pearson 24](#_Toc517886288)

[2.3. *Pertanyaan Penelitian* 25](#_Toc517886289)

[BAB III METODE PENELITIAN 26](#_Toc517886290)

[3.1. *Alat dan Bahan* 26](#_Toc517886291)

[3.1.1. Alat 26](#_Toc517886292)

[3.1.2. Bahan 29](#_Toc517886293)

[3.2. Alur Penelitian 29](#_Toc517886294)

[3.2.1. Identifikasi Masalah 30](#_Toc517886295)

[3.2.2. Pengembangan Aplikasi 31](#_Toc517886296)

[3.2.3. Pengumpulan Data Responden 32](#_Toc517886297)

[3.2.4. Uji Korelasi Data Penggunaan Aplikasi 33](#_Toc517886298)

[3.2.5. Penulisan Laporan 33](#_Toc517886299)

[3.3. Perancangan *Sprint* 33](#_Toc517886300)

[3.4. Prosedur Analisis Uji Korelasi 37](#_Toc517886301)

[3.4.1. Identifikasi Masalah dan Studi Literatur 38](#_Toc517886302)

[3.4.2. Analisis Kebutuhan Data 39](#_Toc517886303)

[3.4.3. Pengambilan Data 40](#_Toc517886304)

[3.4.4. Pengolahan Data 40](#_Toc517886305)

[3.4.5. Analisis Uji Korelasi Pearson 42](#_Toc517886306)

[BAB IV HASIL DAN PEMBAHASAN 43](#_Toc517886307)

[4.1. *Pengembangan Aplikasi* Usage Statistic Recorder 43](#_Toc517886308)

[4.1.1. *Sprint 1* : Layanan Pencatat Data Penggunaan Aplikasi 44](#_Toc517886309)

[4.1.2. *Sprint 2* : Fitur Pengisian Kuesioner VARK 49](#_Toc517886310)

[4.1.3. *Sprint 3* : Persetujuan Pengguna, Petunjuk Penggunaan Aplikasi, dan Pengisian Identitas Responden 56](#_Toc517886311)

[4.1.4. *Sprint 4* : Sinkronisasi Data dengan Firebase 60](#_Toc517886312)

[DAFTAR PUSTAKA 71](#_Toc517886313)

[LAMPIRAN 134](#_Toc517886314)

DAFTAR TABEL

[Tabel 2.1. Tabel perbandingan beberapa sistem posisi untuk lokalisasi [26]. 10](#_Toc509163640)

[Tabel 2.2. Spesifikasi perangkat Google Poject Tango[32]*.* **Error! Bookmark not defined.**](#_Toc509163641)

[Tabel 2.3. Tabel perbandingan fitur SDK tahun 2017 [53]. 26](#_Toc509163642)

[Tabel 3.1. Spesifikasi perangkat keras yang digunakan pada penelitian 34](#_Toc509163651)

[Tabel 3.2. Daftar ponsel pintar pengujian aplikasi 35](#_Toc509163652)

[Tabel 4. 1. Hasil pengujian fungsionalitas dan non-fungsionalitas aplikasi 102](#_Toc509175164)

[Tabel 4. 2. Hasil pengujian fungsionalitas dan non-fungsionalitas aplikasi (Lanjutan) 103](#_Toc509175165)

[Tabel 4. 3. Pengujian keberhasilan pelacakan terhadap intensitas cahaya dan jarak 129](#_Toc509175166)

[Tabel 4. 4. Hasil pengujian aplikasi terhadap beberapa perangkat android 131](#_Toc509175167)

[Tabel 4. 5. Hasil pengujian aplikasi terhadap beberapa perangkat Android (Lanjutan) 132](#_Toc509175168)

DAFTAR GAMBAR

[Gambar 1.1. Contoh peta bangunan pusat perbelanjaan Google Indoor Maps [6]. **Error! Bookmark not defined.**](#_Toc509163682)

[Gambar 2.1. Diagram *fish bone* penelitian mengenai navigasi di dalam bangunan **Error! Bookmark not defined.**](#_Toc509777523)

[Gambar 2.2. *Taxonomy* metode *indoor localization* [18]. **Error! Bookmark not defined.**](#_Toc509777524)

[Gambar 2.3. Ilustrasi RSSI *fingerprinting* [35]*.* 14](#_Toc509777525)

[Gambar 2.4. Prosedur *fingerprinting* [34]. 15](#_Toc509777526)

[Gambar 2.5. Diagram alir algoritma KNN 16](#_Toc509777527)

[Gambar 2.6. Contoh *marker-oriented* berupa QR Code. 19](#_Toc509777528)

[Gambar 2.7. Contoh *markerless* menggunakan metode SLAM. 19](#_Toc509777529)

[Gambar 2.8. Pemindaian lingkungan menggunakan SLAM [61] 21](file:///D:\SKRIPSI\Naskah%20Skripsi\Naskah%20Skripsi%20versi%204.docx#_Toc509777530)

[Gambar 2. 9. Tumpukan perangkat lunak Android [48]. 24](#_Toc509777531)

[Gambar 2.10. Ilustrasi alur struktur data Firebase [50]. 25](#_Toc509777532)

[Gambar 3. 1. Lux meter yang digunakan dalam pengujian aplikasi. 35](#_Toc511102393)

[Gambar 3.2. Diagram alir penelitian. 39](#_Toc511102394)

[Gambar 3.3. Diagram alir pengembangan aplikasi. 42](#_Toc511102395)

[Gambar 3.4. Blok kebutuhan sistem 45](#_Toc511102396)

[Gambar 3.5. Alur proses kerja aplikasi pengunjung acara 46](#_Toc511102397)

[Gambar 3.6. Alur proses kerja aplikasi *event organizer* 47](#_Toc511102398)

[Gambar 3.7 Diagram *use case* aplikasi AR navigasi ruangan 49](#_Toc511102399)

[Gambar 3. 8. *Sequence* diagram aplikasi AR navigasi ruangan pengunjung acara 50](#_Toc511102400)

[Gambar 3. 9. *Sequence* diagram aplikasi AR navigasi ruangan *event organizer* 51](#_Toc511102401)

[Gambar 3. 10. *Activity* diagram aplikasi AR navigasi ruangan pengunjung acara bagian 1 52](#_Toc511102402)

[Gambar 3. 11. *Activity* diagram aplikasi AR navigasi ruangan pengunjung acara bagian 2 53](#_Toc511102403)

[Gambar 3. 12. *Activity* diagram aplikasi AR navigasi ruangan *event organizer* bagian 1 54](#_Toc511102404)

[Gambar 3. 13. *Activity* diagram aplikasi AR navigasi ruangan *event organizer* bagian 2 55](#_Toc511102405)

[Gambar 3. 14. *Activity* diagram aplikasi AR navigasi ruangan *event organizer* bagian 3 56](#_Toc511102406)

[Gambar 3. 15. *Activity* diagram aplikasi AR navigasi ruangan *event organizer* bagian 4 57](#_Toc511102407)

[Gambar 3. 16. Rancangan antarmuka *Low Fidelity Wireframe* aplikasi: a) Antarmuka *sign in* EO; b) Antarmuka utama EO; c) Antarmuka daftar peta EO 58](#_Toc511102408)

[Gambar 3. 17. Rancangan antarmuka *Low Fidelity Wireframe* aplikasi: a) Antarmuka daftar posisi EO; b) Antarmuka *tracking* EO 59](#_Toc511102409)

[Gambar 3. 18. Rancangan antarmuka *Low Fidelity Wireframe* aplikasi: a) Antarmuka pemilihan peta tujuan EO; b) Antarmuka navigasi EO 59](#_Toc511102410)

[Gambar 3. 19. Rancangan antarmuka *Low Fidelity Wireframe* aplikasi: a) Antarmuka fitur mengundang EO; b) Antarmuka *sign in* pengunjung acara; c) Antarmuka pengunjung acara mendapatkan undangan 60](#_Toc511102411)

[Gambar 3. 20. Rancangan antarmuka *Low Fidelity Wireframe* aplikasi: a) Antarmuka *Way to Locate* pengunjung acara; b) Antarmuka navigasi pengunjung acara 61](#_Toc511102412)

[Gambar 3. 21. Rancangan antarmuka *High Fidelity Wireframe* aplikasi: a) Antarmuka *sign in* EO; b) Antarmuka utama EO; c) Antarmuka daftar peta EO 61](#_Toc511102413)

[Gambar 3. 22. Rancangan antarmuka *High Fidelity Wireframe* aplikasi: a) Antarmuka menambah peta baru EO; b) Antarmuka daftar posisi EO; c) Antarmuka menambah posisi baru EO 62](#_Toc511102414)

[Gambar 3. 23. Rancangan antarmuka *High Fidelity Wireframe* aplikasi: a) Antarmuka SLAM EO; b) Antarmuka menambah informasi arah EO 62](#_Toc511102415)

[Gambar 3. 24. Rancangan antarmuka *High Fidelity Wireframe* aplikasi: a) Antarmuka memilih peta tujuan EO; b) Antarmuka navigasi EO 63](#_Toc511102416)

[Gambar 3. 25. Rancangan antarmuka *High Fidelity Wireframe* aplikasi: a) Antarmuka undangan EO; b) Antarmuka *sign in* pengunjung acara 65](#_Toc511102417)

[Gambar 4. 1. Gambaran kerja prototipe aplikasi 66](#_Toc512485439)

[Gambar 4. 2. Fungsi untuk menambahkan *item* peta: a) Tampilan halaman *Map*; b) Tampilan halaman *Position* 68](#_Toc512485440)

[Gambar 4. 3. Tampilan halaman *Wifi Manager* saat menambahkan wifi 69](#_Toc512485441)

[Gambar 4. 4. Titik-titik fitur yang dihasilkan saat proses komputasi SLAM 70](#_Toc512485442)

[Gambar 4. 5. Kode aplikasi untuk memunculkan titik-titik fitur dan menghentikan sementara tititk-titik fitur 70](#_Toc512485443)

[Gambar 4. 6. Hasil objek trimatra koordinat *cartesian* 71](#_Toc512485444)

[Gambar 4. 7. Kode aplikasi untuk memulai proses SLAM dan menghentikan proses SLAM 71](#_Toc512485445)

[Gambar 4. 8. Proses pelacakan SLAM yang gagal 72](#_Toc512485446)

[Gambar 4. 9. Titik-titik fitur ketika berada di ruangan gelap (10 lux) 73](#_Toc512485447)

[Gambar 4. 10. Lokasi penyimpanan hasil SLAM di memori internal : a) Folder penyimpanan berkas; b) File .3dmap hasil proses SLAM 74](#_Toc512485448)

[Gambar 4. 11. Lokasi penyimpanan hasil SLAM di Firebase *storage*: a) Folder penyimpanan berkas; b) File .3dmap hasil proses SLAM 75](#_Toc512485449)

[Gambar 4. 12. Kode Aplikasi untuk menyimpan proses SLAM menjadi .3dmap 76](#_Toc512485450)

[Gambar 4. 13. Bentuk basis data pada Firebase: a) Basis data peta; b) Isi dari data posisi berupa data akses poin; c) Isi dari data posisi berupa data nilai dari RSSI *fingerprinting* 80](#_Toc512485451)

[Gambar 4. 14. Contoh anak panah yang digunakan sebagai *texture* objek trimatra 81](#_Toc512485452)

[Gambar 4. 15. Kode aplikasi *class* *ObjectTrackeRenderer*: a) Kode untuk membuat permukaan objek; b) Kode untuk memberikan *texture* objek 84](#_Toc512485453)

[Gambar 4. 16. Contoh kode aplikasi *class BaseModel* 85](#_Toc512485454)

[Gambar 4. 17. Kode aplikasi *class TexturedCube*: a) Kode untuk membuat objek persegi; b) Kode untuk memberikan *texture*; c) Kode untuk memasukkan *texture* *bitmap* ke dalam objek 88](#_Toc512485455)

[Gambar 4. 18. Perhitungan mundur proses WKNN 88](#_Toc512485456)

[Gambar 4. 19. Objek tirmata berupa anak panah pada layar prianti pintar 89](#_Toc512485457)

[Gambar 4. 20. Kode aplikasi untuk memberikan nilai jenis arah panah 89](#_Toc512485458)

[Gambar 4. 21. Kode aplikasi untuk memuat *file* .3dmap dan menjalankan pada layar ponsel pintar 90](#_Toc512485459)

[Gambar 4. 22. Peringatan untuk menggeser kamera ke arah kanan 90](#_Toc512485460)

[Gambar 4. 23. Kode aplikasi untuk mangatur sudut pandang pengguna: a) Kode fungsi kompas; b) Kode fungsi pengkondisionalan sudut pandang 94](#_Toc512485461)

[Gambar 4. 24. Contoh kode aplikasi fitur *invitation* 95](#_Toc512485462)

[Gambar 4. 25. Contoh kode aplikasi *class DeepLinkManager* 95](#_Toc512485463)

[Gambar 4. 26. Tampilan fitur *invitation*: a) Tampilan untuk memasukkan konten surel; b) Tampilan untuk mengirimkan surel 96](#_Toc512485464)

[Gambar 4. 27. Tampilan *Indoor Navigation Service*: a) Bentuk *icon* aplikasi b) Tampilan *launcher*; c) Tampilan autentikasi EO; d) tampilan autentikasi pengunjung acara; e) Tampilan *sign in*; f) Tampilan *register* 99](#_Toc512485465)

[Gambar 4. 28. Tampilan halaman utama EO 99](#_Toc512485466)

[Gambar 4. 29. Tampilan non-fungsionalitas: a) Tampilan menu; b) Tampilan instruksi; c) Tampilan tentang aplikasi 102](#_Toc512485467)

[Gambar 4. 30. Undangan yang dikirim oleh EO: a) Undangan diterima lewat surel; b) Undangan diterima lewat SMS 105](#_Toc512485468)

[Gambar 4. 31. Aktivitas saat *link* pada undangan ditekan: a) *Link* diarahkan ke Play Store; b) *Link* diarahkan ke aplikasi 106](#_Toc512485469)

[Gambar 4. 32. Tampilan setelah pengguna memberi izin akses splikasi: a) Tampilan *invitation* *id*; b) Tampilan autentikasi aplikasi *attendy* 107](#_Toc512485470)

[Gambar 4. 33. Tampilan aplikasi *attendy*: a) Peringatan jika pengguna belum terdaftar; b) Tampilan utama aplikasi *attendy* 108](#_Toc512485471)

[Gambar 4. 34. Aplikasi *attendy*: a) Daftar acara yang diundang; b) Daftar peta navigasi tujuan yang tersedia 108](#_Toc512485472)

[Gambar 4. 35. Tampilan navigasi AR: a) Proses perhitungan WKNN; b) Objek trimatra tertampil pada layar ponsel pintar 109](#_Toc512485473)

[Gambar 4. 36. Informasi berupa orientasi sudut untuk membantu sudut pandang kamera 110](#_Toc512485474)

[Gambar 4. 37. Kinerja aplikasi *Indoor Navigation Sevice*: a) Rata-rata penggunaan memori perangkat; b) Detail penggunaan kinerja aplikasi 110](#_Toc512485475)

[Gambar 4. 38. Pengujian skenario di laboratorium informatika 112](#_Toc512485476)

[Gambar 4. 39. Luaran ketika hanya 1 akses poin yang dipilih 113](#_Toc512485477)

[Gambar 4. 40. Luaran ketika lebih dari 2 akses poin yang dipilih 113](#_Toc512485478)

[Gambar 4. 41. Percobaan pengujian akurasi aplikasi menggunakan 3 akses poin 114](#_Toc512485479)

[Gambar 4. 42. Grafik persentase akurasi penentuan posisi terhadap jumlah akses poin yang digunakan dari 30 kali percobaan 114](#_Toc512485480)

[Gambar 4. 43. Ilustrasi pengujian pada gedung bertingkat 116](#_Toc512485481)

[Gambar 4. 44. Luaran yang salah pada pengujian di lantai 2 116](#_Toc512485482)

[Gambar 4. 45. Luaran yang salah pada pengujian koridor lantai 2 117](#_Toc512485483)

[Gambar 4. 46. Proses SLAM pada ruangan dengan intensitas cahaya paling rendah (0 lux) 119](#_Toc512485484)

[Gambar 4. 47. Proses SLAM pada ruangan dengan intensitas cahaya tinggi (250 lux) 119](#_Toc512485485)

[Gambar 4. 48. Proses SLAM pada ruangan dengan intensitas cahaya 20 lux 121](#_Toc512485486)

[Gambar 4. 49. Proses SLAM pada ruangan dengan intensitas cahaya 100 lux 121](#_Toc512485487)

[Gambar 4. 50. Proses SLAM yang dipengaruhi jumlah objek tertampil (intensitas 200 lux) 122](#_Toc512485488)

[Gambar 4. 51. Kamera ponsel pintar ketika mendapat gangguan 122](#_Toc512485489)

[Gambar 4. 52. Kamera ponsel pintar dapat melakukan pelacakan tanpa penanda kembali 123](#_Toc512485490)

[Gambar 4. 53. Ilustrasi jangkauan sudut pandang pengguna terhadap ruangan yang terlacak 124](#_Toc512485491)

[Gambar 4. 54. Pengujian sudut pandang: a) Pengujian sudut pandang 00; b) Pengujian sudut pandang 150 125](#_Toc512485492)

[Gambar 4. 55. Pengujian sudut pandang: a) Pengujian sudut pandang 450; b) Pengujian sudut pandang 900 125](#_Toc512485493)

[Gambar 4. 56. Ilustrasi sudut pandang *eye level,* *low level*, *high level* 126](#_Toc512485494)

[Gambar 4. 57. Pengujian sudut pandang *low level*: a) Sudut pandang sejajar dengan mata; b) Sudut pandang di bawah 10 cm dari mata 126](#_Toc512485495)

[Gambar 4. 58. Pengujian sudut pandang *low level*: a) Sudut pandang di bawah 30 cm dari mata; b) Sudut pandang di bawah 60 cm dari mata 126](#_Toc512485496)

[Gambar 4. 59. Pengujian sudut pandang *high level*: a) Sudut pandang sejajar dengan mata; b) Sudut pandang di atas 10 cm dari mata 127](#_Toc512485497)

[Gambar 4. 60. Pengujian sudut pandang *high level*: a) Sudut pandang di atas 30 cm dari mata; b) Sudut pandang di atas 60 cm dari mata 127](#_Toc512485498)

[Gambar 4. 61. Hasil pengujian jarak pada intensitas cahaya 50 lux 130](#_Toc512485499)

[Gambar 4. 62. Hasil pengujian jarak pada intensitas cahaya 200 lux 130](#_Toc512485500)

[Gambar 4. 63. Aplikasi gagal dipasang pada ponsel Sony Xperia L C2105 133](#_Toc512485501)

[Gambar 4. 64. Percobaan pertama ponsel Lava Iris 870 4G 134](#_Toc512485502)

[Gambar 4. 65. Percobaan kedua ponsel Lava Iris 870 4G 135](#_Toc512485503)

DAFTAR SINGKATAN

**2**

2D Dua Dimensi

**3**

3D Tiga Dimensi

**A**

AP *Access Point*

API *Application Programming Interface*

**B**BaaS *Backend as a Service*

BSSID *Basic Service Set Identifier*

**C**CAD *Computer Aided Design*

CRUD *Create Read Update Delete*

**D**dB *Desibel*

**G**GPS *Global Positioning System*

**I**IPS *Indoor Positioning System*

**J**JSON *JavaScript Object Notation*

**K**KNN *K Nearest Neighbors*

**L**LBS *Location Based Service*

**M**

MAC *Media Access Control*

**N**NoSQL *Not only Structured Query Language*

**O**OpenGL *Open Graphics Library*

OCR *Optical Character Recognition*

**Q**QR *Quick Response*

**R**RAD *Rapid Application Development*

RFID *Radio Frequency Identification*

RSSI *Received Signal Strength Indicator*

**S**SDK *Standard Development Kit*

SLAM *Simultaneous Localization and Mapping*

SSID *Service Set Identifier*

**U**UI *User Interface*

UML *Unified Modelling Language*

**W**Wifi *Wireless Fidelity*

WKNN *Weighted K Nearest Neighbors Algorithm*

**X**XML *Extensible Markup Language*

INTISARI

Bangunan saat ini tersusun dari banyak lantai, ruangan, dan koridor. Terkadang hal tersebut membuat seseorang merasa kesulitan dalam mencari dan menuju sebuah lokasi di dalam bangunan. Kebanyakan seseorang memiliki pengalaman tersesat ketika melakukan navigasi di koridor, pusat perbelanjaan, atau bangunan yang memiliki bentuk bangunan yang terlihat identik. Setiap orang memiliki keterampilan, pengalaman, dan interpretasi yang berbeda dalam memanfaatkan informasi seperti tanda atau petunjuk informasi menjadi sebuah rute navigasi.

Keterbatasan GPS dalam mendapatkan koordinat lokasi di dalam ruangan karena sinyal satelit yang terblokir. Salah satu metode untuk melakukan navigasi di dalam bangunan melalui penggabungan metode *indoor localization* berbasis *Augmented Reality* (AR). Ciri khas teknologi AR mampu menampilkan visualisasi objek trimatra sebagai informasi navigasi. Hal tersebut dapat menutupi kelemahan dari metode *indoor localization* yaitu orientasi sudut pandang. Penelitian ini mengusulkan kerangka kerja aplikasi untuk aplikasi berbasis Android dengan menggabungkan metode RSSI *fingerprinting* sebagai lokalisasi posisi dan pelacakan tanpa penanda pada ruang tertutup (*indoor*) untuk navigasi.

Hasil pengujian diperoleh bahwa aplikasi dapat diterapkan di dalam bangunan dengan memperhatikan beberapa faktor, meliputi kondisi pencahayaan ruang, sudut pandang, jarak, akurasi serta jenis perangkat bergerak yang digunakan. Akurasi posisi pada penggunaan 3 akses poin adalah 30% dan akurasi maksimal adalah 100% pada 10 akses poin.

**Kata kunci :** *augmented reality*, pemetaan simultan dan lokalisasi, RSSI *fingerprinting*, *indoor localization*, visualisasi

*ABSTRACT*

*Currently, building is composed of many floors, rooms, and corridors. Sometimes it makes person difficult to find and go to location inside the building. Most of people have lost experience when navigating in corridors, shopping malls, or buildings that looks identical. Everyone has different skills, experiences, and interpretations in utilizing information such as information from marks or guides into a navigation route.*

*The limitations of GPS in getting the coordinates of the location indoors due to blocked satellite signals. One of methods to navigate inside building is through the incorporation of indoor localization method based Augmented Reality (AR). The characteristic of AR technology is capable of displaying visualization of augmented objects as navigation information. It can cover the weakness of indoor localization method that is point of view. This research proposes an application framework for Android apps by incorporating RSSI fingerprinting method as location locality and markerless tracking in indoor space to display navigation.*

*From the results, application can be applied in the building by taking into several factors, including the light conditions, point of views, distance, accuracy and types of mobile devices used.* *Position accuracy on the use of 3 access points is 30% and maximum accuracy is 100% at 10 access points.*

***Keywords :*** *augmented reality, simultaneous localization and mapping, RSSI fingerprinting, indoor localization, visualization*

BAB I   
PENDAHULUAN

## *Latar Belakang*

Kegiatan sehari-hari manusia dewasa ini tidak lepas dari pengaruh dan penggunaan teknologi informasi, maka kualitas tampilan informasi pada suatu teknologi informasi menjadi suatu hal yang penting untuk diperhatikan. Salah satu masalah yang dapat mempengaruhi kualitas tampilan informasi adalah *cognitive overload* [1]. Masalah tersebut muncul karena terlalu banyak informasi yang ditampilkan bagi pengguna, sehingga pengguna justru mengalami kesulitan dalam memproses informasi yang hendak disampaikan. Solusi dari masalah ini adalah dengan menampilkan informasi dalam bentuk yang sesuai dengan karakteristik pengguna. Salah satu karakteristik yang dapat diamati adalah gaya belajar pengguna, karena pemrosesan informasi berkaitan erat dengan kegiatan belajar.

Gaya belajar merupakan kecenderungan cara seseorang dalam mengolah informasi untuk menambah pengetahuan atau keterampilan baru [2]. Gaya belajar memiliki 4 dimensi, yaitu kognitif, afektif, fisiologis, dan psikologis [3]. Secara konvensional, gaya belajar dapat diidentifikasi secara manual dengan menggunakan kuesioner [4, 5], akan tetapi metode tersebut memiliki kelemahan dalam aspek akurasi [6]. Hal tersebut mendorong sejumlah peneliti untuk mengembangkan pendekatan otomatis untuk mengidentifikasi gaya belajar [6, 7, 8]. Pendekatan otomatis dilakukan dengan mengumpulkan data responden yang diperlukan, untuk kemudian diolah menggunakan algoritma-algoritma kecerdasan buatan atau diujikan terhadap aturan-aturan yang telah ditentukan oleh pakar. Tahapan awal dalam pengembangan sistem pengidentifikasi gaya belajar otomatis adalah menentukan media pengamatan dan model gaya belajar yang akan dijadikan acuan. Model gaya belajar akan menjadi dasar dalam menentukan 2 komponen utama dalam pengembangan sistem, yaitu fitur dan pola perilaku [9]. Fitur merupakan konten atau materi pembelajaran yang tersedia pada media pengamatan. Pola perilaku merupakan suatu pola yang terbentuk dari interaksi pengguna terhadap fitur-fitur yang tersedia. Kedua komponen tersebut kemudian diklasifikasikan menurut karakteristik dari setiap model pada permodelan gaya belajar acuan. Terdapat sejumlah model gaya belajar yang telah dikembangkan, antara lain model *Felder-Silverman Learning Style Model* (FSLSM) [4], model Kolb [10], dan VARK (*Visual, Aural, Read/Write, Kinesthetic*) [5]. Model FSLSM dan Kolb berfokus pada aspek kognitif dan psikologis, sedangkan VARK berfokus pada aspek fisiologis [3]. VARK mengklasifikasikan responden ke dalam 4 kategori berdasarkan preferensinya terhadap format materi pembelajaran yang diakses. Keempat format tersebut adalah visual, aural, tekstual, dan kinestetik. Solusi yang berusaha dikembangkan pada penelitian ini berkaitan erat dengan bentuk tampilan informasi, suatu aspek yang sangat erat dengan aspek fisiologis dalam pembelajaran. Berdasarkan penjabaran tersebut, maka model gaya belajar VARK menjadi model yang paling relevan dan akan digunakan dalam penelitian ini. Sudah ada penelitian yang menggunakan pendekatan otomatis dan model gaya belajar VARK [7], namun terdapat sejumlah kelemahan pada penelitian tersebut. Kelemahan tersebut terletak pada bagian media pengamatan dan pola perilaku yang diamati.

Kelemahan pada media pengamatan berkaitan dengan proses pembelajaran yang terjadi di dalamnya yang bersifat non-formal [11]. Terkait hal tersebut, kegiatan belajar manusia secara keseluruhan dapat dikategorikan ke dalam 3 bentuk, yaitu formal, non-formal, dan informal [11]. Kategori yang paling dominan dalam kehidupan manusia adalah pembelajaran informal. Pembelajaran informal berperan terhadap 75% dari keseluruhan pengetahuan yang dimiliki manusia [12]. Berdasarkan teori tersebut, dapat dikatakan bahwa pembelajaran daring sebagai media pengamatan hanya mampu menggambarkan sebagian kecil dari keseluruhan proses pembelajaran yang dilakukan manusia dengan porsi sebesar-besarnya 25%. Penelitian ini berusaha menangani kelemahan tersebut dengan memilih media pengamatan yang mendukung proses pembelajaran informal. Pada dasarnya, pembelajaran informal dilakukan manusia melalui setiap pengalaman pada kehidupan sehari-hari, seperti pekerjaan rumah tangga, internet, perangkat-perangkat elektronik, bahkan melalui percakapan biasa sehari-hari [11]. Berdasarkan kriteria-kriteria yang dibutuhkan, *smartphone* dapat menjadi media pengamatan yang ideal. *Smartphone* sangat dekat dengan proses pembelajaran informal, karena media komunikasi elektronik ini sangat dekat dengan kehidupan manusia sehari-hari di zaman sekarang. Media ini juga memiliki fitur-fitur pengamatan yang sesuai dengan model gaya belajar VARK. Hasibuan *et.al* mengklasifikan fitur-fitur pengamatannya berdasarkan format-format multimedia, sebagai contoh fitur visual meliputi materi-materi dengan format flv, mp4, dan dat [7]. Fitur-fitur multimedia juga tersedia pada *smartphone* dalam bentuk aplikasi, sebagai contoh aplikasi Youtube dapat diklasifikasikan sebagai fitur visual. Hal tersebut menunjukkan bahwa *smartphone* layak untuk dijadikan sebagai media pengamatan. Fitur yang akan diamati adalah aplikasi-aplikasi yang terdapat pada *smartphone* pengguna.

Kelemahan kedua terletak pada pola perilaku yang diamati. Hasibuan *et.al* hanya mengamati 1 pola perilaku saja, yaitu durasi akses pengguna terhadap fitur-fitur yang tersedi [7]. Pada penelitian lain, terdapat pola perilaku tambahan yang dapat diamati, yaitu frekuensi akses pengguna terhadap fitur-fitur yang tersedia. Pola perilaku tersebut layak untuk diamati, karena penelitian yang mengamati pola perilaku tersebut berhasil mencapai tingkat akurasi yang tinggi, yaitu 80.7% [6]. Akan tetapi, model gaya belajar yang digunakan oleh penelitian tersebut kurang cocok dengan permasalahan pada penelitian ini. Oleh karena itu, penelitian ini akan berusaha mengkombinasikan kedua penelitian di atas. Model gaya belajar dan fitur yang diamati akan mengacu pada penelitian pertama, sedangkan pola perilaku akan ditambahkan dari penelitian kedua. Dengan demikian, parameter-parameter yang relevan dalam pengklasifikasian kecenderungan gaya belajar telah teridentifikasi. Akan tetapi, belum ada penelitian yang menyatakan secara eksplisit bahwa parameter-parameter tersebut memiliki korelasi yang signifikan terhadap pengklasifikasian kecenderungan gaya belajar dalam lingkungan pembelajaran informal*,* maka perlu dilakukan uji korelasi.

## *Rumusan Masalah*

Terdapat 4 aspek utama yang perlu diperhatikan dalam pengembangan sistem pengidentifikasi gaya belajar otomatis, yaitu model gaya belajar, media pengamatan, fitur, dan pola perilaku. Sejumlah penelitian telah dilakukan untuk mengidentifikasi gaya belajar melalui pendekatan otomatis, akan tetapi penelitian-penelitian tersebut memiliki kelemahan pada aspek media pengamatan. Penelitian-penelitian tersebut menggunakan media pengamatan yang hanya mendukung proses pembelajaran non-formal saja. Padahal, jenis kegiatan belajar yang paling dominan dalam kehidupan manusia adalah pembelajaran informal. Pembelajaran informal berperan terhadap 75% dari keseluruhan pengetahuan yang dimiliki manusia. Oleh karena itu, pengamatan terhadap pembelajaran non-formal dapat dikatakan kurang mampu menggambarkan keseluruhan proses pembelajaran yang dialami manusia selama hidupnya.

Pembelajaran informal lebih baik dalam menggambarkan proses pembelajaran yang dilakukan manusia dalam kehidupan sehari-hari. Salah satu media pengamatan yang sangat dekat dengan sebagian besar proses pembelajaran informal manusia sehari-hari di zaman sekarang adalah *smartphone*. Media pengamatan ini juga memiliki kesesuaian fitur dengan penelitian sebelumnya yang telah berhasil dilakukan menggunakan media pembelajaran daring [7]. Dengan mengacu pada penelitian tersebut, dapat ditentukan parameter-parameter pengamatan berupa frekuensi dan durasi penggunaan aplikasi pada *smatphone* dengan pengklasifikasian aplikasi berdasarkan model gaya belajar VARK. Akan tetapi, belum ada penelitian yang menyatakan secara eksplisit bahwa parameter-parameter tersebut memiliki korelasi yang signifikan terhadap pengklasifikasian kecenderungan gaya belajar dalam lingkungan pembelajaran informal.

## *Keaslian Penelitian*

Keaslian dari penelitian ini akan dibahas berdasarkan 4 aspek utama dalam pengembangan sistem pengidentifikasi gaya belajar otomatis, yaitu model gaya belajar, media pengamatan, fitur, dan pola perilaku. Perbandingan akan dilakukan terhadap 2 penelitian sebelumnya yang dilakukan oleh Bernard *et. al* [6] dan Hasibuan *et.al* [7].

Model gaya belajar yang akan digunakan dalam penelitian ini adalah model gaya belajar VARK [5]. Model ini digunakan karena memiliki dimensi pengamatan yang sesuai untuk menyelesaikan permasalahan *cognitive overload*, yaitu dimensi fisiologis. Sedangkan model gaya belajar lain seperti FSLSM [4] dan Kolb [10] berfokus pada dimensi kognitif dan psikologis [3]. Bernard *et. al* menggunakan model FSLSM, sedangkan Hasibuan *et. al* menggunakan model VARK.

Media pengamatan yang digunakan dalam penelitian ini adalah *smartphone*, sedangkan kedua penelitian sebelumnya menggunakan media pembelajaran daring. Penelitian ini menggunakan *smartphone* untuk mengatasi kelemahan yang terdapat pada media pembelajaran daring sebagai media pengamatan. Media pembelajaran daring hanya mendukung proses pembelajaran non-formal dalam pengamatannya. Proses pembelajaran non-formal berperan jauh lebih sedikit dibandingkan proses pembelajaran informal dalam kehidupan manusia. Proses pembelajaran informal berperan terhadap 75% terhadap keseluruhan pengetahuan yang dimiliki manusia [12], yang berarti proses pembelajaran non-formal hanya memiliki porsi peran sebesar-besarnya 25%. Berdasarkan kriteria pembelajaran informal, media pengamatan berkaitan erat dengan proses pembelajaran informal adalah *smartphone*.

Fitur-fitur pengamatan yang digunakan dalam penelitian ini mengacu pada fitur yang digunakan Hasibuan *et.al* dalam pembelajaran daring, namun disesuaikan dengan skenario penelitian ini yang menggunakan *smartphone* sebagai media pengamatan. Hasibuan *et.al* mengklasifikan fitur-fitur pengamatannya berdasarkan format-format multimedia, sebagai contoh fitur visual meliputi materi-materi dengan format flv, mp4, dan dat [7]. Fitur-fitur multimedia juga tersedia pada *smartphone* dalam bentuk aplikasi, sebagai contoh aplikasi Youtube dapat diklasifikasikan sebagai fitur visual. Maka, fitur yang akan diamati pada penelitian ini adalah aplikasi-aplikasi yang terdapat pada *smartphone* pengguna. Bernard *et. al* di lain sisi, menggunakan fitur-fitur yang lebih berfokus pada dimensi kognitif pada media pembelajaran daring, yaitu ringkasan, konten, contoh, soal ujian, soal latihan, evaluasi mandiri, dan forum.

Pola perilaku yang akan diamati pada penelitian ini adalah frekuensi dan durasi akses terhadap fitur-fitur pengamatan. Pola perilaku ini dipilih dengan mengacu pada pola perilaku yang digunakan Bernard *et. al*, namun diterapkan bagi fitur-fitur yang lebih berfokus pada dimensi fisiologis pelajar. Hal ini dilakukan untuk melengkapi kekurangan pola perilaku yang diamati oleh Hasibuan *et.al* yang hanya mengamati durasi akses saja. Adapun gambaran keseluruhan dari perbandingan antara penelitian ini dengan kedua penelitian acuan lainnya dapat dilihat pada Tabel 1.1.

Tabel 1.1. Tabel perbandingan aspek-aspek penelitian

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Penelitian** | **Model Gaya Belajar** | **Media Pengamatan** | **Fitur** | **Pola Perilaku** |
| M S Hasibuan *et. al* | VARK | *E-learning* (non-formal) | Materi pembelajaran dalam format flv, mp4, dat, mp3, acc, jurnal, *e-book*, dan modul. | Durasi akses |
| Bernard *et. al* | FSLSM | *E-learning* (non-formal) | Ringkasan | Frekuensi akses |
| Konten |
| Contoh |
| Soal ujian |
| Soal latihan | Durasi akses |
| Evaluasi mandiri |
| Forum |
| Penelitian ini | VARK | *Smartphone* (informal) | Kegunaan dari aplikasi-aplikasi *smartphone* yang relevan terhadap VARK | Frekuensi akses |
| Durasi akses |

## *Batasan Masalah*

Untuk menyelesaikan permasalahan tersebut, penelitian ini akan berfokus dengan batasan masalah berikut:

1. Penelitian ini melakukan uji korelasi antara peubah frekuensi dan durasi penggunaan aplikasi *smartphone* terhadap peubah gaya belajar VARK. Penelitian ini tidak sampai pada tahap identifikasi/pengelompokkan seseorang ke dalam gaya belajar VARK.
2. Penelitian ini dilakukan pada *smartphone* dengan sistem operasi Android.
3. Sistem perekam penggunaan aplikasi yang dikembangkan memastikan keaktifan pengguna dalam menggunakan aplikasi dengan mendeteksi kondisi *foreground*/*background* dari aplikasi yang direkam. Pengguna dianggap aktif pada aplikasi tertentu ketika aplikasi berada dalam kondisi *foreground.*

## *Tujuan Penelitian*

Tujuan yang akan dicapai dalam penelitian ini adalah:

1. Mengembangkan *service* berbasis Android untuk menangkap peubah-peubah yang akan digunakan dalam uji korelasi. Parameter-parameter tersebut adalah data frekuensi dan durasi penggunaan aplikasi pada *smartphone* berbasis Android.
2. Melakukan uji korelasi antara frekuensi dan durasi penggunaan aplikasi *smartphone* terhadap gaya belajar VARK.

## *Manfaat Penelitian*

Manfaat yang diharapkan dari penelitian ini adalah memberikan gambaran parameter-parameter mana saja yang dapat digunakan untuk mengembangkan sistem pendeteksi gaya belajar otomatis melalui pendekatan pembelajaran informal berbasis *smartphone*, sehingga dapat menjadi dasar bagi pemecahan masalah *cognitive overload* pada tampilan sistem informasi.

## *Sistematika Penulisan*

Penulisan laporan hasil penelitian disusun dalam lima bab dengan penjelasan sebagai berikut:

1. BAB I: PENDAHULUAN

Bab ini berisi tentang latar belakang penelitian dilakukan, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, keaslian penelitian, dan sistematika penulisan laporan penelitian.

1. BAB II: TINJAUAN PUSTAKA DAN DASAR TEORI

Bab ini berisi tinjauan pustaka dari penelitian terkait dengan topik deteksi gaya belajar, kekurangan dari penelitian sebelumnya, serta potensi dilakukannya penelitian ini. Bab ini juga berisi teori-teori dasar tentang belajar, model gaya belajar VARK, *smartphone*, aplikasi piranti bergerak, android, *App Usage Statistic API*, basis data Firebase, metode pengembangan aplikasi SCRUM, dan teori uji korelasi Pearson

1. BAB III: METODE PENELITIAN

Bab ini membahas alat dan bahan yang digunakan dalam penelitian ini. Selain itu, bab ini juga menjelaskan alur penelitian dari awal hingga akhir yang dilakukan oleh penulis.

1. BAB IV: HASIL DAN PEMBAHASAN

Bab ini membahas hasil penelitian yang telah dilakukan, termasuk hasil aplikasi yang dikembangkan serta hasil analisis korelasi data penggunaan aplikasi *smartphone* terhadap gaya belajar VARK.

1. BAB V: KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan akhir dari penelitian dan analisis data yang telah dilakukan. Bab ini juga berisi saran untuk penelitian lebih lanjut.

BAB II  
TINJAUAN PUSTAKA DAN DASAR TEORI

## *Tinjauan Pustaka*

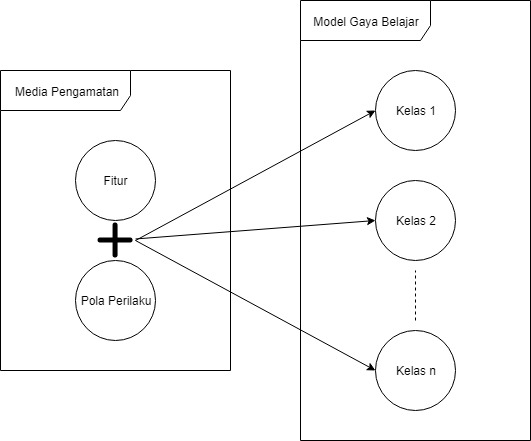
Pengidentifikasian gaya belajar dapat dilakukan menggunakan pendekatan manual maupun otomatis. Pendekatan manual menggunakan kuisioner untuk mengidentifikasi gaya belajar responden [4, 5], akan tetapi pendekatan manual memiliki kelemahan dalam hal akurasi. Kelemahan tersebut muncul karena kuisioner menarik responden keluar dari aktivitas normal mereka. Hal ini menciptakan kesadaran pada responden akan maksut dari kuisioner tersebut, sehingga dapat menimbulkan bias pada objektivitas jawaban responden [6]. Sejumlah peneliti mencoba menangani kelemahan tersebut dengan menerapkan pendekatan otomatis dalam mengidentifikasi gaya belajar seseorang.

Graf menyatakan bahwa pendekatan otomatis dapat dilakukan dengan dua metode yang berbeda, yaitu metode *data-driven* dan metode berbasis literatur [13]. Metode *data-driven* menggunakan masukan berupa data-data perilaku seseorang yang kemudian diolah dengan algoritma-algoritma dari bidang kecerdasan buatan (AI). Metode berbasis literatur menggunakan aturan-aturan yang sudah ditentukan oleh pakar, berkaitan dengan interaksi user terhadap bahan-bahan ajar yang tersedia. Aturan-aturan ini kemudian digunakan untuk memprediksi gaya belajar pengguna. Penelitian ini mencakupi tahapan awal pada metode *data-driven*, yaitu mengumpulkan data masukan mengenai perilaku subjek identifikasi.

Pham *et.al* memaparkan metodologi untuk melakukan pengidentifikasian gaya belajar dengan pendekatan otomatis [8]. Terdapat 4 aspek utama yang perlu ditentukan dalam mengembangkan sistem pengidentifikasi gaya belajar otomatis, yaitu model gaya belajar acuan, media pengamatan, fitur, dan pola perilaku. Model gaya belajar acuan menjadi aspek yang paling penting dan mendasar, karena aspek-aspek lainnya akan ditentukan berdasar model gaya belajar acuan. Terdapat berbagai model gaya belajar, antara lain model FSLSM, model Kolb, dan VARK. Model FSLSM dan Kolb lebih berfokus pada aspek kognitif dan psikologis, sedangkan VARK lebih berfokus pada aspek fisiologis [3]. Penelitian ini terkait dengan bentuk tampilan informasi, suatu aspek yang sangat erat dengan aspek fisiologis dalam pembelajaran. Oleh sebab itu, model gaya belajar VARK menjadi model yang paling relevan bagi penelitian ini. Aspek selanjutnya yang perlu ditentukan adalah media pengamatan. Media pengamatan yang paling populer untuk digunakan dalam pengidentifikasian gaya belajar otomatis adalah pembelajaran daring. Beberapa penelitian sebelumnya [6, 7, 8] menggunakan media pengamatan pembelajaran daring. Media pengamatan berkaitan erat dengan bentuk kegiatan belajar yang terjadi di dalamnya.

Radakovi mengkategorikan kegiatan belajar manusia ke dalam 3 bentuk, yaitu formal, non-formal, dan informal [11]. Pembelajaran informal merupakan suatu bentuk kegiatan belajar yang paling berpengaruh terhadap pengetahuan yang dimiliki manusia, karena 75% dari keseluruhan pengetahuan yang dimiliki manusia didapatkan melalui pembelajaran informal [12]. Pembelajaran daring termasuk ke dalam kategori pembelajaran non-formal [11], sehingga dikatakan bahwa pembelajaran daring sebagai media pengamatan hanya mampu menggambarkan kurang dari 25% dari keseluruhan proses pembelajaran yang dilakukan subjek penelitian. Penelitian ini berusaha mengatasi permasalahan tersebut dengan menggunakan media pengamatan berbentuk pembelajaran informal. Tahapan yang perlu dilakukan setelah menentukan model gaya belajar acuan dan media pengamatan adalah menentukan fitur dan pola perilaku yang akan diamati.

Fitur merupakan konten yang tersedia pada media pengamatan. Pola perilaku merupakan pola yang terbentuk dari interaksi pengguna terhadap fitur-fitur yang tersedia. Sejumlah kombinasi dari fitur dan pola perilaku akan dikelompokkan sesuai kelas-kelas yang telah didefinisikan oleh model gaya belajar acuan. Sebagai contoh, Hasibuan *et.al* menggunakan e-learning sebagai media pengamatan dan VARK sebagai permodelan gaya belajar acuan [7]. Fitur yang digunakan adalah materi-materi pembelajaran yang tersedia dalam berbagai format multimedia. Pola perilaku yang diamati adalah durasi pengaksesan tiap fitur. Salah satu contoh pengelompokan kombinasi fitur dan pola perilaku adalah sebagai berikut, durasi pengaksesan fitur dengan format mp3 akan masuk dalam kelompok aural (A) dari VARK. Penelitian ini sebenarnya masih dapat dioptimalkan dengan menambahkan pola perilaku yang diamati, yaitu frekuensi pengaksesan fitur. Bernard *et.al* telah menggunakan pola perilaku tersebut dan terbukti memiliki tingkat akurasi hasil yang tinggi [6]. Hanya saja penelitian tersebut menggunakan model gaya belajar FSLSM yang kurang relevan bagi penelitian ini. Oleh sebab itu, penelitian ini berusaha mengkombinasikan kedua penelitian di atas dengan menerapkan pola perilaku yang digunakan penelitian Bernard *et.al* [6] terhadap model gaya belajar acuan yang digunakan penelitian Hasibuan *et.al* [7].



Gambar 2.1.Relasi antara komponen-komponen penelitian

Berdasarkan ulasan di atas dan studi literatur secara keseluruhan, secara garis besar dapat digambarkan relasi antar komponen seperti yang ditampilkan pada Gambar 2.1. Media pengamatan terdiri dari berbagai konten. Konten-konten tersebut kemudian diseleksi untuk mencari konten-konten yang relevan bagi penelitian, sehingga menghasilkan fitur-fitur yang akan diamati. Pengguna dapat melakukan sejumlah interaksi terhadap fitur-fitur tersebut. Interaksi yang akan diamati dinamakan pola perilaku. Kombinasi antara fitur dan pola perilaku kemudian dikelompokkan ke dalam kelas-kelas yang ada pada model gaya belajar acuan.

## *Dasar Teori*

### Belajar

Belajar sering didefinisikan sebagai perubahan perilaku yang terjadi akibat bertambahnya pengetahuan [14, 15], akan tetapi anggapan ini dianggap memiliki keterbatasan. Keterbatasan itu muncul karena definisi ini seakan-akan mengabaikan faktor-faktor lain yang dapat mengakibatkan perubahan perilaku, seperti motivasi dan kondisi fisik [16, 17]. Selain itu, belajar juga sering didefinisikan sebagai bertambahnya pengetahuan melalui suatu kondisi yang secara sengaja telah direncanakan dalam lingkungan formal [18, 19]. Definisi ini juga memiliki keterbatasan, karena penelitian menunjukkan bahwa proses pembelajaran terjadi setiap saat, baik secara disengaja maupun tidak [19]. Maka, belajar lebih cocok jika didefinisikan sebagai segala bentuk pertambahan pengetahuan melalui 3 kategori proses pembelajaran, yaitu formal, non-formal, dan informal [11].

Pembelajaran formal adalah pembelajaran yang diberikan oleh institusi pendidikan bagi pelajar dan dilakukan secara terstruktur (dalam hal target-target, waktu, dan dukungan dalam proses pembelajaran). Pembelajaran formal berujung pada sertifikasi pelajar. Pembelajaran formal dilakukan secara sengaja berdasarkan sudut pandang pelajar. Kategori kedua yaitu pembelajaran non-formal, merupakan suatu proses pembelajaran yang diselenggarakan dalam bentuk aktivitas-aktivitas yang terencana yang tidak dirancang sebagai proses pembelajaran secara eksplisit (dalam hal target-target, waktu, dan dukungan dalam proses pembelajaran), namun mengandung elemen-elemen pembelajaran yang penting. Contoh pembelajaran formal antara lain kursus, sekolah, dan segala bentuk pelatihan yang menawarkan sertifikat. Pembelajaran non-formal dilakukan secara sengaja berdasarkan sudut pandang pelajar dan biasanya tidak menawarkan sertifikat. Salah satu contoh pembelajaran non-formal adalah pembelajaran daring terstruktur. Salah satu contoh pembelajaran non-formal adalah pembelajaran daring terstruktur. Kategori ketiga yaitu pembelajaran informal, merupakan pembalajaran yang terjadi selama seumur hidup dalam kehidupan manusia, yang didapatkan melalui kegiatan kegiatan sehari-hari. Pembelajaran ini terjadi tanpa disengaja, maka manusia dapat menambah pengetahuan secara tidak sadar melalui proses pembelajaran ini. Adapun perbandingan dari 3 kategori ini disajikan dalam Tabel 2.1.

Tabel 2.1. Tabel perbandingan kategori proses pembelajaran

|  |  |  |  |
| --- | --- | --- | --- |
| **Kategori** | **Komponen Terstruktur** | **Unsur Kesengajaan** | **Sertifikasi** |
| Formal | Target | Disengaja | Ya |
| Materi |
| Waktu |
| Dukungan |
| Non-formal | Materi | Disengaja | Umumnya tidak |
| Informal | Tidak ada | Tidak disengaja | Tidak |

Belajar dapat dipengaruhi oleh berbagai faktor dan setiap faktor memiliki dampak yang berbeda-beda bagi setiap orang. Hal ini dapat diamati dari gaya belajar mereka. Gaya belajar merupakan kecenderungan cara seseorang dalam mengolah informasi untuk menambah pengetahuan atau keterampilan baru [2]. Gaya belajar memiliki 4 dimensi sebagai berikut [3]:

1. Kognitif: mengamati gaya belajar berdasarkan proses belajar dan bagaimana seseorang mengolah informasi.
2. Afektif: mengamati gaya belajar berdasarkan suasana pikiran dan benak seseorang yang dapat mempengaruhi proses belajar mereka.
3. Fisiologis: mengamati gaya belajar berdasarkan fitur-fitur biologis seseorang, seperti bagaimana kecenderungan seseorang menggunakan indera yang mereka miliki dalam proses belajar.
4. Psikologis: mengamati gaya belajar dengan memperhatikan aspek psikologi dan kepribadian seseorang.

### Model Gaya Belajar VARK

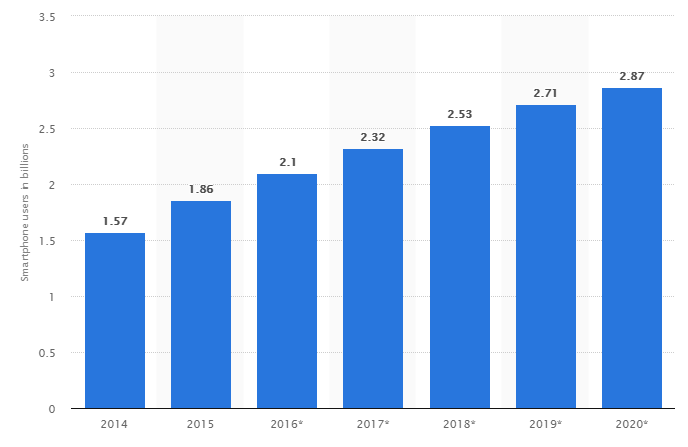
VARK merupakan salah satu model gaya belajar. VARK merupakan singkatan dari *Visual, Aural, Read/Write, and Kinesthetic*. Model ini mengklasifikasikan individu ke dalam empat kategori berdasarkan preferensinya terhadap format bahan ajar yang disampaikan. Keempat format tersebut adalah sebagai berikut [2, 5]:

1. *Visual*: preferensi ini mencakupi bagan, diagram, simbol, dan diagram sebagai representasi dari suatu informasi. Individu dengan preferensi ini mudah terganggu atau teralihkan oleh gerakan atau aksi.
2. *Aural*: preferensi ini mencakupi diskusi berkelompok, penyampaian verbal, e-mail, diskusi daring, dan radio sebagai representasi dari suatu informasi. E-mail yang bersifat tekstual dapat dikategorikan dalam preferensi ini karena kerap mengandung bahasa-bahasa non-formal layaknya percakapan sehari-hari. Individu dengan preferensi ini mudah terganggu atau teralihkan oleh suara. Selain itu, individu dengan preferensi ini memiliki kecenderungan untuk melafalkan bahan bacaan dengan cukup lantang.
3. *Read/Write*: preferensi ini mencakupi segala informasi yang disajikan secara tekstual.
4. *Kinesthetic*: individu dengan preferensi ini cenderung memahami informasi melalui pengalaman, contoh nyata, melakukan praktek, demonstrasi, dan simulasi. Individu dengan preferensi ini cenderung banyak bergerak ketika sedang berfikir.

Preferensi gaya belajar VARK seseorang dapat diidentifikasi melalui metode konvensional menggunakan kuesioner resmi VARK [20]. Kuesioner resmi VARK dikeluarkan oleh lembaga VARK Learn Limited dan dapat diakses melalui situs resmi VARK [20]. Kuesioner VARK terbaru telah mencapai versi 7.1. Tiap pertanyaan dari kuesioner VARK akan memiliki empat pilihan jawaban. Masing-masing jawaban bernilai 1 poin V, A, R, atau K. Responden dipersilahkan memilih pilihan jawaban yang paling sesuai dengan kecenderungan responden dalam menyikapi persoalan sehari-hari. Responden dapat memilih lebih dari 1 jawaban pada pertanyaan yang sama jika memang jawaban-jawaban tersebut menggambarkan keadaan responden. Sebaliknya, responden juga diperbolehkan untuk tidak memilih jawaban sama sekali pada suatu pertanyaan jika di antara jawaban tersebut ternyata tidak ada yang sesuai dengan responden. Kemudian masing-masing pilihan jawaban responden akan dihitung berapa total nilai V, A, R, dan K-nya. Kategori dengan poin tertinggi menggambarkan preferensi gaya belajar responden tersebut, apakah responden termasuk dalam kelompok *Visual, Aural, Read/Write*, atau *Kinesthetic*.

### *Smartphone*

Tidak terdapat definisi pasti dari *smartphone,* namun *smarphone* sering dianggap sebagai komputer berukuran kecil. Anggapan tersebut muncul karena *smartphone* memikiki kemampuan untuk menjalankan berbagai jenis aplikasi layaknya komputer pribadi (PC). Aplikasi yang terdapat pada *smartphone* dapat berasal dari pabrikan atau pun dari toko aplikasi virtual. Dengan dukungan dari aplikasi peranti bergerak, *smartphone* mampu memfasilitasi pengguna dengan berbagai fungsi seperti navigasi, penjelajahan internet, pengolahan kata, penyunting gambar, dll. Kekayaan fungsi *smartphone* telah mendorong popularitas perangkat ini untuk terus berkembang dan semakin mempengaruhi kehidupan manusia. Data dari Statista menunjukkan bahwa jumlah pengguna *smartphone* di dunia terus bertambah setiap tahunnya, bahkan diprediksi akan menyentuh angka 2.87 miliar pengguna di tahun 2020. Adapun grafik jumlah pengguna *smartphone* dapat dilihat pada Gambar 2.2.



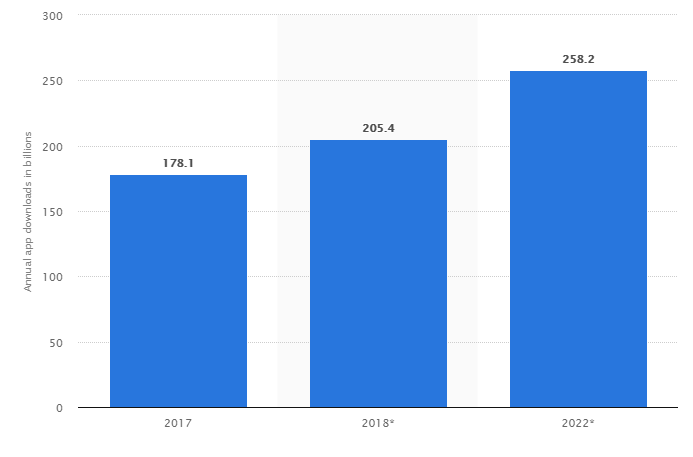
Gambar 2.2.Grafik jumlah pengguna *smartphone* [21]

Garis batas yang tidak jelas membuat orang terkadang sulit membedakan *smartphone* dari telepon genggam konvensional, ataupun peranti komputer bergerak lain seperti tablet. Berdasarkan segi ukuran, *smartphone* umumnya seukuran dengan telepon genggam konvensional dan relatif jauh lebih kecil dari komputer tablet. Secara fungsional, *smartphone* mampu menyediakan fungsi-fungsi seperti yang terdapat pada PC dan komputer tablet. Hal ini membuat *smartphone* lebih “pintar” dari telepon genggam konvensional. Pada dasarnya telepon genggam konvensional hanya mampu menyediakan layanan telepon dan pertukaran pesan tekstual, sedangkan *smartphone* mampu melakukan jauh lebih banyak hal lain dengan dukungan aplikasi peranti bergerak [22].

### Aplikasi Peranti Bergerak

Aplikasi peranti bergerak merupakan suatu tipe aplikasi perangkat lunak yang dirancang khusus untuk bekerja pada peranti bergerak seperti *smartphone* atau komputer tablet [23]. Jenis aplikasi ini menyediakan layanan-layanan yang mirip dengan aplikasi pada PC, akan tetapi aplikasi ini memiliki fungsi-fungsi yang lebih terbatas dan berukuran lebih kecil. Karakteristik tersebut tercipta sebagai bentuk penyesuaian terhadap keterbatasan sumber daya perangkat keras yang terdapat pada peranti bergerak.

Pada awal kemunculannya, aplikasi ini dikembangkan dengan mengadopsi aplikasi berbasis PC yang telah ada ke dalam lingkungan peranti bergerak. Akan tetapi, popularitas peranti bergerak terus bekembang pesat hingga melampaui PC. Permintaan pasar terhadap aplikasi peranti bergerak pun turut meningkat. Data dari Statista menunjukkan bahwa jumlah pengunduhan aplikasi peranti bergerak di tahun 2017 telah mencapai 178 miliar dan diprediksi akan terus meningkat tiap tahunnya, seperti yang nampak pada Gambar 2.3. Hal ini mendorong banyak pengembang perangkat lunak untuk menerapkan pendekatan *mobile first* [24]. Pendekatan ini dilakukan dengan mengembangkan aplikasi secara spesifik bagi lingkungan peranti bergerak sebagai aplikasi utama, barulah dikembangkan versi PC dari aplikasi tersebut. Pengembangan aplikasi peranti bergerak memiliki faktor-faktor pertimbangan yang berbeda dari pengembangan aplikasi berbasis PC. Hal tersebut dikarenakan karakteristik peranti-peranti bergerak yang bergantung pada daya baterai dan memiliki variasi ukuran layar, kapasitas memori, kemampuan prosesor, sensor, dan tipe-tipe interaksi masukan yang lebih beragam.

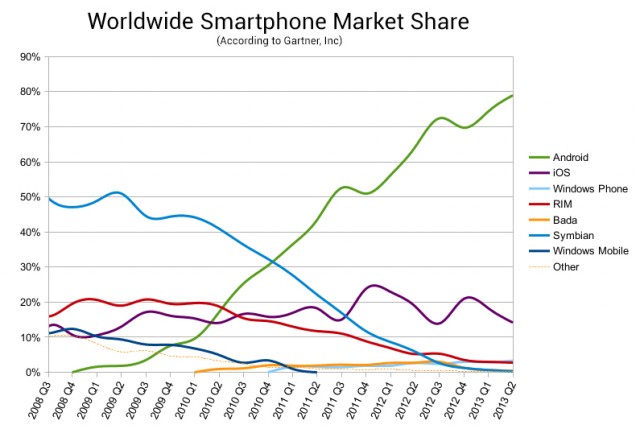


Gambar 2.3.Grafik jumlah pengunduhan aplikasi peranti bergerak [25]

### Android

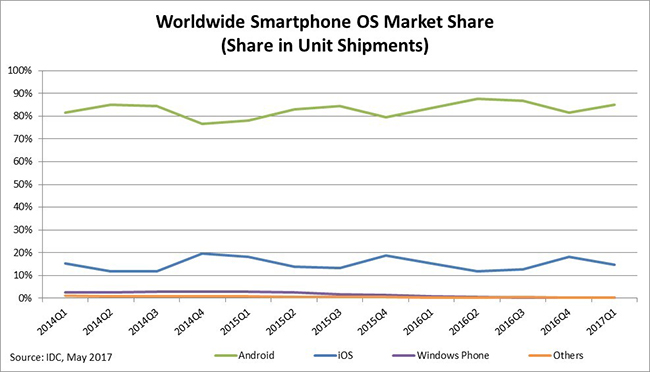
Android merupakan sistem operasi peranti bergerak berbasis Linux. Sistem operasi ini dirancang terutama bagi peranti bergerak dengan layar sentuh. Seiring perkembangannya, kini Android juga dikembangkan bagi televisi, mobil, jam tangan cerdas, dll. Android dikembangkan oleh Google, Android juga terhubung dengan serangkaian perangkat lunak eksklusif milik Google, seperti Gmail, Google Play Store, Google Search, Google Calendar, dan Google Maps. Salah satu ciri khas Android adalah keterbukaannya.

Android bersifat *open source*, sehingga banyak pihak ketiga yang dapat menggunakan *platform* Android secara gratis untuk dikembangkan sesuai kepentingan masing-masing [26]. Google juga membentuk suatu kelompok bernama Open Handset Alliance yang terdiri dari perusahaan-perusahaan perangkat keras, perangkat lunak, dan telekomunikasi untuk turut berkontribusi dalam pengembangan Android. Strategi keterbukaan Android ini membuat Android berkembang dengan sangat cepat dan memberi dampak positif bagi perkembangan pasar Android. Gambar 2.4 menunjukkan bahwa perkembangan pasar Android sangatlah pesat di awal pengembangannya hingga mampu menguasai pasar dengan porsi hampir 80% di kuartal kedua tahun 2013.



Gambar 2.4.Grafik pasar *smartphone* berdasarkan sistem operasi [27]

Setelah itu, Android berhasil menjaga stabilitas pasarnya pada kisaran 80% hingga kuartal pertama tahun 2017 menurut data IDC seperti yang ditampilkan pada Gambar 2.5. Strategi ini memang terbukti berdampak positif bagi perkembangan Android, namun strategi ini juga menimbulkan isu. Isu yang timbul dari masalah ini adalah isu fragmentasi. Keterbukaan Android telah menimbulkan banyaknya variasi perangkat Android, namun tidak semuanya menerapkan versi terbaru dari Android. Hal ini berarti para pengguna perangkat Android yang tidak terbaharui tidak akan mendapatkan akses ke fitur-fitur terbaru dari Android. Isu ini menjadi isu serius karena data menunjukkan bahwa persentase pengguna yang masih menggunakan versi lama dari Android mencapai 30% [28].



Gambar 2.5.Grafik pasar sistem operasi *smartphone* tahun 2014-2017 [29]

### App Usage Statistic API

Mulai versi Lollipop (Android 5.0), Android menyediakan sebuah API (*Aplication Programmable Interface*) bernama App Usage Statistic API. Dengan memanfaatkan API ini, sebuah aplikasi android dapat diprogram untuk mengakses data historis penggunaan aplikasi pada *smartphone* dalam interval waktu tertentu. Untuk menggunakan APP Usage Statistic API, syarat minimal yang harus dipenuhi adalah:

1. Android SDK 27
2. Android Build Tools v27.0.2
3. Android Support Repository

Interval waktu yang disediakan oleh API ini adalah INTERVAL\_DAILY untuk rentang harian, INTERVAL\_MONTHLY untuk rentang bulanan, INTERVAL\_YEARLY untuk rentang tahunan, dan INTERVAL\_BEST untuk menentukan rentang waktu secara otomatis [30]. Untuk masing-masing interval tersebut, diperlukan parameter waktu mulai (*begin time*) dan waktu selesai (*end time*). API akan memberikan hasil (*result*) berupa daftar aplikasi yang dibuka oleh *smartphone* pada rentang waktu tersebut.

Gambar 2.6 berikut ini adalah contoh parameter yang diinput ke dalam API dan hasil yang diberikan oleh API tersebut:

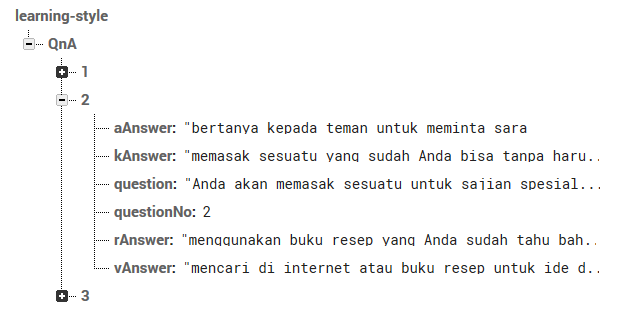
|  |
| --- |
| intervalType = INTERVAL\_YEARLY beginTime = 2013 endTime = 2015 (exclusive)   Results:  2013 - com.example.alpha  2013 - com.example.beta  2014 - com.example.alpha  2014 - com.example.beta  2014 - com.example.charlie |

Gambar 2.6. Contoh hasil masukan dan keluaran *App Usage Statistic* API

Bukan hanya itu saja, API ini dapat memberikan durasi penggunaan aplikasi selama rentang waktu tersebut. Untuk mengakses durasi penggunaan aplikasi, maka fungsi yang dipanggil adalah getTotalTimeInForeground(). Sebuah aplikasi dapat dikatakan aktif (pengguna melakukan interaksi terhadap aplikasi tersebut) jika aplikasi tersebut berada di depan layar (*foreground*), bukan di balik layar (*background*). Selain durasi penggunaan aplikasi, API ini juga mampu memberikan frekuensi penggunaan aplikasi dengan cara menghitung total berapa kali aplikasi tersebut berpindah dari *background* ke *foreground.*

### Firebase

Basis data adalah sebuah kumpulan data. Atau dalam pengertian yang lain basis data adalah cara / metode / teknologi untuk menyimpan data. Secara umum basis data dibagi menjadi dua kategori, yaitu basis data terstruktur (Structured Query Language / SQL) dan basis data tidak terstruktur (NoSQL) [31]. Basis data terstruktur menyimpan data dalam bentuk tabel yang terdiri dari kolom dan baris. Tabel-tabel dalam suatu basis data dapat memiliki relasi terhadap tabel lainnya. Artinya sebuah informasi yang dibutuhkan oleh pengguna atau aplikasi bisa diperoleh atau disusun dari beberapa tabel yang berbeda.



Gambar 2.6.Contoh basis data tidak terstruktur

Firebase adalah sebuah sistem basis data NoSQL yang dikembangkan oleh Google. Berbeda dengan basis data SQL, basis data NoSQL tidak dibatasi oleh tabel dan relasi antar tabel. Data disimpan dalam pasangan kunci dan nilai (*key and value*). Pasangan kunci dan nilai dapat memiliki pasangan kunci dan nilai di dalamnya, atau bisa disebut sebagai anak (*child*). Jika digambarkan secara visual, pasangan-pasangan kunci dan nilai tersebut akan membentuk sebuah skema yang mirip dengan sebuah pohon seperti pada Gambar 2.6. Maka format basis data firebase sering disebut sebagai sebuah JSON Tree. JSON sendiri merupakan singkatan dari Java Script Object Notation, yaitu sebuah format pertukaran data berbasis bahasa pemrograman Java Script. Namun karena kemudahan penggunaan dan independensinya, JSON juga dapat digunakan dalam berbagai bahasa pemrograman lain, seperti C, C++, C#, JavaScript, Perl, Python, dan lainnya [32, 33]. Selain itu, Firebase memiliki kelebihan dibandingkan basis data tidak terstruktur lainnya, yaitu Firebase mendukung layanan basis data secara *real time*.

### Evernote Android Job

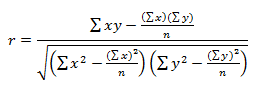
Sebuah *service* adalah potongan program atau instruksi pada sebuah aplikasi Android yang bekerja di balik layar. *Service* diperlukan karena tidak semua proses komputasi yang dilakukan oleh Android harus ditampilkan ke pengguna dalam bentuk antarmuka / *interface*, misalnya proses baca tulis data dari basis data, proses sinkronisasi data dengan *server*, dan sebagainya. Evernote Android Job adalah sebuah pustaka (*library*) yang memberikan kemudahan dalam membuat sebuah *service* / *background process* di Android. Pembuatan sebuah *service* di Android memiliki kendala utama yaitu metode pembuatan *service* berbeda-beda tergantung dari versi sistem operasi Android yang terpasang. Dengan menggunakan pustaka ini, maka pengembang aplikasi Android tidak perlu lagi membuat *service* yang berbeda untuk tiap versi sistem operasi Android. Cukup dengan membuat sebuah *service* atau *Job*, maka secara otomatis *Job* tersebut dapat berjalan di semua versi sistem operasi Android.

### Metode Pengembangan Perangkat Lunak SCRUM

Terdapat berbagai metode untuk mengembangkan sebuah sistem atau sebuah perangkat lunak. SCRUM [34] adalah salah satu metode pengembangan perangkat lunak berbasis iterasi atau pengulangan. Iterasi dalam SCRUM disebut sebagai *sprint*. Pengembangan aplikasi dibagi ke dalam beberapa *sprint*, di mana setiap akhir sebuah *sprint* harus terjadi penambahan nilai atau fitur yang nyata ke dalam aplikasi. Metode SCRUM terdiri dari beberapa tahapan, yaitu Product Backlog, Sprint Backlog, Sprint, Sprint Review dan Product Increment [34].

### Uji Korelasi Pearson

Tingkat ketergantungan/dependensi dari 2 peubah x dan y dapat dihitung untuk menghasilkan suatu nilai ketergantungan. Nilai ketergantungan tersebut disebut sebagai koefisien korelasi. Koefisien korelas antara dua peubah x dan y dapat dihitung dengan rumus [35]:

 (1)

dimana:

r : nilai korelasi antara peubah x dan y

x : nilai peubah pertama

y : nilai peubah kedua

n : banyaknya sampel

Nilai r selalu terletak di antara rentang -1 < r < 1 [35]. Jika nilai r = 0, maka kedua peubah x dan y sama sekali tidak memiliki korelasi. Atau dengan kata lain kedua peubah x dan y bersifat independen, tidak saling mempengaruhi satu sama lain. Jika nilai r = 1, maka kedua peubah x dan y memiliki korelasi sempurna dan bersifat sebanding. Artinya, semakin tinggi nilai x, maka semakin tinggi pula nilai y, dan sebaliknya. Jika nilai r = -1, maka kedua peubah x dan y memiliki korelasi sempurna dan bersifat berbanding terbalik. Artinya, semakin tinggi nilai x, maka semakin rendah nilai y, dan sebaliknya. Sedangkan untuk nilai-nilai r selain nilai 0, -1, dan 1, korelasi atau nilai r dapat dikategorikan ke dalam tiga kelompok, yaitu *small* (0,1 - 0,3), *medium* (0,3 - 0,5) dan *large* (> 0,5) [36]. Hal ini berlaku baik untuk r bernilai positif maupun r bernilai negatif.

## *Pertanyaan Penelitian*

1. Apakah layanan yang dikembangkan dapat menangkap data durasi dan frekuensi penggunaan aplikasi?
2. Apakah data durasi penggunaan aplikasi berkorelasi terhadap gaya belajar VARK?
3. Apakah data frekuensi penggunaan aplikasi berkorelasi terhadap gaya belajar VARK?

BAB III  
METODE PENELITIAN

## *Alat dan Bahan*

Beberapa alat dan bahan yang digunakan dalam penelitian ini adalah sebagai berikut:

### Alat

Alat yang digunakan dalam penelitian ini dibagi menjadi dua bagian, yaitu perangkat keras dan perangkat lunak sebagai berikut:

* 1. Perangkat Keras

Penelitian dilakukan dengan menggunakan 2 laptop dengan spesifikasi seperti pada Tabel 3.1 dan Tabel 3.2, serta 2 *smartphone* dengan spesifikasi seperti pada Tabel 3.3 dan Tabel 3.4.

Tabel 3.1. Spesifikasi laptop 1 yang digunakan dalam penelitian

|  |  |
| --- | --- |
| Seri | Asus A455L |
| CPU | Intel (R) Core(TM) i5-4210U CPU @ 1.70 GHz |
| GPU | NVIDIA GeForce 820M |
| Resolusi Layar | 1366 x 768 |
| RAM | 4.00 GB |
| HDD | 500 GB |
| OS | Windows 8 Enterprise 64-bit |
| Linux Ubuntu 16.04 LTS |

Tabel 3.2. Spesifikasi laptop 2 yang digunakan dalam penelitian

|  |  |
| --- | --- |
| Seri | Asus K401LB |
| CPU | Intel(R) Core(TM) i5-5200U CPU @ 2.20GHz (4 CPUs), ~2.2GHz |
| GPU | NVIDIA GeForce 940M |
| Resolusi Layar | 1920 x 1080 |
| RAM | 8.00 GB |
| HDD | 1 TB |
| OS | Windows 8.1 Pro 64-bit (6.3, Build 9600) |

Tabel 3.3. Spesifikasi *smartphone* 1 yang digunakan dalam penelitian

|  |  |
| --- | --- |
| Seri | Xiaomi Redmi 3 |
| Chipset | Qualcomm MSM8939v2 Snapdragon 616 |
| Prosesor | Octa-core (4x1.5 GHz Cortex-A53 & 4x1.2 GHz Cortex-A53) |
| GPU | Adreno 405 |
| Memori | 16 GB |
| RAM | 2 GB |
| OS | Android 5.1 (Lollipop) |
| Versi MIUI | MIUI Global 7.0 |

Tabel 3.4. Spesifikasi *smartphone* 2 yang digunakan dalam penelitian

|  |  |
| --- | --- |
| Seri | Xiaomi Redmi 3 |
| Chipset | Qualcomm MSM8916 Snapdragon 615 |
| Prosesor | Octa-core (4x1.5 GHz Cortex-A53 & 4x1.2 GHz Cortex-A53) |
| GPU | Adreno 405 |
| Memori | 32 GB |
| RAM | 3 GB |
| OS | Android 5.1 (Lollipop) |
| Versi MIUI | MIUI Global 7.9 |

* 1. Perangkat Lunak

Beberapa perangkat lunak yang digunakan untuk mengembangkan aplikasi adalah:

1. Android Studio

Android Studio adalah sebuah *Integrated Development Environtment* (IDE) atau lingkungan pengembangan yang terintegrasi untuk mengembangkan aplikasi berbasis Android. Android Studio merupakan IDE resmi yang dikembangkan oleh Google dengan dasar bahasa pemrograman Java untuk memprogram logika aplikasi dan bahasa *Ekstensible Markup Language* (XML) untuk memprogram sumber-sumber daya data, seperti tampilan, warna, teks, dan sebagainya. Android Studio yang digunakan dalam penelitian ini adalah Android Studio versi 3.1.

1. Evernote Job Library

*Evernote Job Library* adalah sebuah pustaka tambahan untuk Android Studio. Pustaka ini membantu mengembangkan layanan yang berjalan di latar belakang suatu smartphone (*background service*) atau disebut sebagai Android Job. Fitur yang dimanfaatkan dari pustaka ini adalah *Job Builder*, *Run Job Immediately*, dan *Schedule Periodic Job*.

1. Firebase

Firebase yang digunakan dalam aplikasi ini adalah versi 11.2.0. Fitur-fitur Firebase yang dimanfaatkan adalah *Realtime Database* sebagai basis data tak terstruktur dan *Authentication* sebagai sarana autentikasi pengguna. Fitur tersebut memudahkan pengembangan aplikasi karena mampu menyederhanakan fungsi *backend server* sehingga proses pengembangan dapat berfokus pada pengembangan aplikasi itu sendiri.

1. Microsoft Excel

Microsoft Excel adalah aplikasi *spreadsheet* untuk pengaturan, analisis, dan penyimpanan data dalam bentuk tabel. Versi yang akan digunakan dalam penelitian ini adalah versi 16. Perangkat lunak ini akan digunakan untuk melakukan pengolahan dan visualisasi data. Pengolahan yang dilakukan adalah pembersihan dan normalisasi data.

1. SPSS

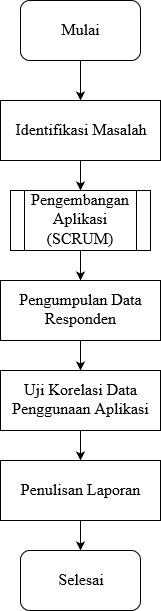
SPSS merupakan perangkat lunak yang digunakan untuk mendukung analisis statistika. SPSS akan digunakan pada penelitian ini sebagai alat bantu dalam melakukan analisis uji korelasi.

### Bahan

Bahan yang digunakan dalam penelitian ini berupa data dan informasi pendukung yang berasal dari buku, jurnal, makalah, thesis, dan *international conference* yang terkait. Selain itu bahan yang akan diolah dalam penelitian ini adalah hasil pengisian kuesioner VARK dari responden.

## Alur Penelitian

Penelitian dimulai dengan identifikasi permasalahan pada topik identifikasi gaya belajar serta peluang penelitian yang dapat dilakukan. Identifikasi masalah dilakukan dengan studi literatur baik secara daring maupun luring. Hasil dari studi literatur digunakan sebagai landasan penelitian dan penentuan permasalahan utama yang akan diselesaikan serta batasan-batasan penelitian. Kemudian, aplikasi dikembangkan dengan metode SCRUM. Aplikasi ini bertujuan untuk menangkap data durasi dan frekuensi penggunaan *smartphone* milik responden. Kemudian data tersebut akan dilakukan uji korelasi dan dianalisis hasilnya. Tahap terakhir dari penelitian ini adalah penulisan laporan. Alur penelitian secara lengkap dari awal sampai akhir dapat digambarkan melalui Gambar 3.1.



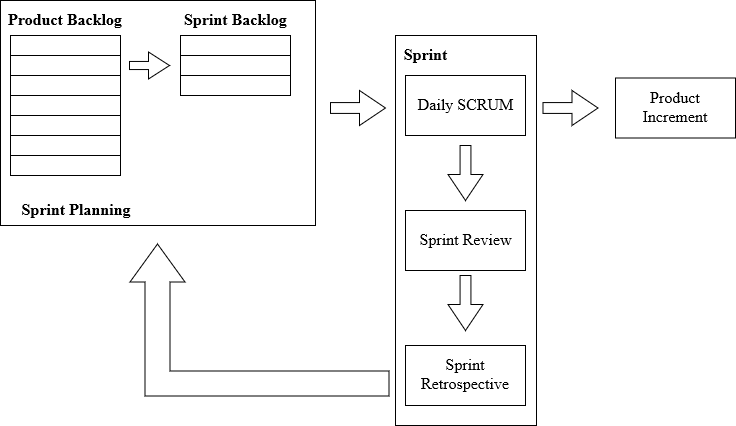
Gambar 3.1. Diagram alir penelitian

## Identifikasi Masalah

Dilakukan identifikasi permasalahan-permasalahan yang melatarbelakangi penelitian pada tahap ini. Bukan hanya itu saja, kemungkinan solusi untuk menyelesaikan permasalahan tersebut juga dicari dari berbagai referensi yang ada. Studi literatur yang dilakukan mencakup pencarian teori yang relevan serta penelitian-penelitian yang pernah dilakukan pada bidang permasalahan tersebut untuk mempekuat permasalahan dan mendasari penelitian ini dilakukan. Referensi yang digunakan bersumber dari buku, jurnal, artikel, dan situs-situs internet.

## Pengembangan Aplikasi

Aplikasi dikembangkan dengan kerangka kerja SCRUM. Tahapan pengembangan aplikasi dengan metode SCRUM dibagi menjadi beberapa tahap seperti pada Gambar 3.2.



Gambar 3.2.Metode pengembangan aplikasi SCRUM

1. Product Backlog

Tahap *Product Backlog* merupakan tahap analisis secara lengkap tentang kebutuhan aplikasi yang akan dibangun. Dengan kata lain tahap ini adalah tahap membuat kumpulan hal-hal atau fitur-fitur yang harus tersedia dalam aplikasi yang akan dibangun.

1. Sprint Backlog

Tahap *Sprint Backlog* adalah tahap awal sebelum dilakukan *Sprint*, yaitu memilih satu atau beberapa *Product* *Backlog* yang akan dikembangkan dalam *Sprint* berikutnya. Tahap ini juga akan menentukan durasi dari *Sprint* yang akan dilaksanakan. Durasi *Sprint* berkisar antara 1-4 minggu.

1. Daily Scrum

Sebuah *Sprint* dibagi lagi ke dalam proses pengembangan harian (*daily*). Di awal hari, akan ditentukan apa yang akan dilakukan atau dikembangkan dalam satu hari ke depan.

1. Sprint Review

Akhir sebuah *Sprint* adalah tahap *Sprint Review*, yaitu menijau ulang kembali apa saja yang sudah diselesaikan / tercapai dan apa saja yang belum dalam *Sprint* yang telah dilakukan.

1. Sprint Restrospective

Pada akhir sebuah *Sprint* juga dilakukan *Sprint Restrospective*. Jika *Sprint Review* berfokus pada bahasan fitur-fitur yang sudah dikerjakan (*what*), maka tahap *Sprint Restrospective* berfokus pada bagaimana cara *Sprint* sebelumnya bekerja (*how*). Dilakukan peninjauan tentang perlu atau tidaknya perbaikan dalam cara kerja selama *Sprint* berlangsung.

1. Product Increment

*Product Increment* merupakan penambahan nilai / fitur / fungsional aplikasi pada akhir sebuah *Sprint*. Penambahan tersebut tentu saja harus mengacu pada *Sprint backlog* dan *Product backlog* yang telah disepakati pada awal *Sprint*.

## Pengumpulan Data Responden

Setelah aplikasi selesai dikembangkan, tahap penelitian berikutnya adalah pengumpulan data responden. Dua data yang berbeda dikumpulkan menggunakan media aplikasi yang telah dikembangkan. Data pertama yang dikumpulkan adalah data preferensi gaya belajar pengguna berdasarkan hasil pengisian kuesioner VARK. Data tersebut berupa total nilai untuk masing-masing kategori V, A, R, dan K. Data kedua yang dikumpulkan adalah data penggunaan aplikasi pada *smartphone* responden. Data tersebut berupa durasi dan frekuensi penggunaan aplikasi-aplikasi yang ada pada *smartphone* tersebut. Selain itu, pada tahap ini juga dikumpulkan data identitas dari responden berupa nama lengkap, usia, dan jenis kelamin.

## Uji Korelasi Data Penggunaan Aplikasi

Setelah data berhasil dikumpulkan dari responden, tahap berikutnya dari penelitian ini adalah melakukan uji korelasi antara data penggunaan aplikasi terhadap gaya belajar responden. Uji korelasi bertujuan untuk mengetahui apakah ada keterkaitan antara gaya belajar seorang responden terhadap cara responden berinteraksi dengan *smartphone* milik responden.

## Penulisan Laporan

Tahap terakhir dari penelitian ini adalah penulisan laporan. Tahap ini akan menguraikan secara rinci permasalahan yang melatarbelakangi peneilitan dilakukan, tujuan penelitian, metode yang digunakan, serta hasil dari penelitian ini.

## Perancangan *Sprint*

Pada metode pengembangan aplikasi SCRUM, tahap awal yang dilaksanakan adalah perancangan *sprint* yang terdiri dari penyusunan *product backlog* dan *sprint backlog*. Untuk menyusun *product backlog* dan *sprint* *backlog*, dibutuhkan analisis fitur-fitur aplikasi dan kebutuhan-kebutuhan aplikasi, baik kebutuhan secara fungsionalitas maupun non-fungsionalitas seperti pada Tabel 3.5. Analisis kebutuhan ini diperlukan untuk memberikan gambaran tentang aplikasi yang akan dibuat beserta fitur-fiturnya.

Tabel 3.5. Fitur dan kebutuhan aplikasi

| **Fitur** | **Detail** | **Kebutuhan** |
| --- | --- | --- |
| Kuesioner VARK | Responden mengisi kuesioner gaya belajar VARK melalui aplikasi | Aplikasi mampu menampilkan 16 pertanyaan dari kuesioner VARK beserta 4 pilihan jawaban |
| Pengguna dapat memilih lebih dari satu jawaban atau tidak memilih sama sekali |
| Aplikasi mampu menampilkan urutan pertanyaan kuesioner secara acak |
| Aplikasi mampu menampilkan urutan pilihan jawaban pertanyaan kuesioner secara acak |
| Daftar pertanyaan dan jawaban berasal dari basis data Firebase |
| Aplikasi mampu menghitung poin untuk masing-masing kategori V,A,R, dan K |
| Layanan Penangkap Data Penggunaan Aplikasi | Layanan untuk menangkap data durasi dan frekuensi penggunaan aplikasi-aplikasi pada *smartphone* responden | Layanan dapat menangkap data historis durasi penggunaan aplikasi |
| Layanan dapat menangkap data historis frekuensi penggunaan aplikasi |
| Layanan harus dapat berjalan di balik layar (*background*) |
| Autentikasi, Penyimpanan Data, Persetujuan dan Petunjuk Penggunaan | Autentikasi untuk mengetahui pengguna sudah mengisi kuesioner / belum. Penyimpanan data untuk menyimpan data hasil kuesioner dan penggunaan aplikasi ke Firebase | Aplikasi mampu menampilkan persetujuan pengguna dan petunjuk penggunaan aplikasi |
| Hanya pengguna yang seutju untuk berpartisipasi yang dapat menggunakan aplikasi |
| Pengguna harus memasukkan data pengguna seperti nama, email, jenis kelamin, usia, dan pendidikan terakhir |
| Aplikasi mampu menampilkan pesan jika data yang dimasukkan pengguna belum lengkap |
| Hanya pengguna yang belum mengisi kuesioner yang dapat melakukan pengisian kuesioner |
| Aplikasi mampu menyimpan data identitas pengguna, hasil kuesioner dan penggunaan aplikasi ke Firebase |
| Penyelesaian akhir aplikasi | Fitur-fitur yang telah dibuat digabungkan menjadi satu aplikasi utuh dan disusun berdasarkan alur kerja aplikasi | Pembuatan alur kerja aplikasi |
| Penyusunan urutan fitur-fitur aplikasi berdasarkan alur kerja aplikasi |

Setelah identifikasi fitur dan kebutuhan aplikasi dilakukan, langkah penelitian berikutnya adalah penyusunan *product backlog* berdasarkan fitur aplikasi seperti pada Tabel 3.6. Penyusunan *product backlog* dilengkapi dengan penyusunan skala prioritas fitur yang diperlukan oleh aplikasi.

Tabel 3.6.*Product Backlog* aplikasi

|  |  |
| --- | --- |
| **Prioritas** | **Product Backlog** |
| 1 | Layanan Penangkap Data Penggunaan Aplikasi |
| 2 | Kuesioner VARK |
| 3 | Persetujuan Pengguna, Petunjuk Penggunaan Aplikasi, dan Pengisian Identitas Responden |
| 4 | Sinkronisasi Firebase, Autentikasi, dan Penyelesaian Akhir |

Berikutnya, *product backlog* tersebut dijabarkan secara lebih rinci ke dalam beberapa *sprint backlog* seperti pada Tabel 3.7 untuk layanan penangkap data penggunaan aplikasi, Tabel 3.8 untuk kuesioner VARK, Tabel 3.9 untuk persetujuan pengguna dan petunjuk penggunaan aplikasi, serta Tabel 3.10 untuk sinkronisasi Firebase dan autentikasi.

Tabel 3.7. *Sprint* 1: layanan penangkap data penggunaan aplikasi

|  |  |
| --- | --- |
| **Prioritas** | **Product Backlog** |
| **1** | **Layanan Penangkap Data Penggunaan Aplikasi** |
|  | Membuat menu akses pintas ke pengaturan perangkat bagian *App Usage Access* |
|  | Mengembangkan layanan penangkap data durasi penggunaan aplikasi |
|  | Mengembangkan layanan penangkap data frekuensi penggunaan aplikasi |
|  | Membuat layanan berjalan pada *background* |
|  | *Sprint overview* |

Tabel 3.8. *Sprint* 2 : kuesioner VARK

|  |  |
| --- | --- |
| **Prioritas** | **Product Backlog** |
| **2** | **Kuesioner VARK** |
|  | Mempersiapkan daftar pertanyaan dan jawaban kuesioner serta menterjemahkannya ke dalam bahasa Indonesia |
|  | Memasukkan daftar pertanyaan dan jawaban kuesioner ke dalam Firebase |
|  | Mengembangkan tampilan Android pengisian kuesioner VARK |
|  | Mengembangkan metode untuk mengacak urutan pertanyaan kuesioner |
|  | Mengembangkan metode untuk mengacak urutan jawaban tiap pertanyaan kuesioner |
|  | Mengembangkan metode untuk menampilkan pertanyaan dan kuesioner |
|  | Mengembangkan metode untuk menghitung hasil kuesioner |
|  | Mengembangkan halaman akhir |
|  | *Sprint overview* |

Tabel 3.9. *Sprint* 3 : persetujuan pengguna, petunjuk penggunaan aplikasi dan pengisian identitas responden

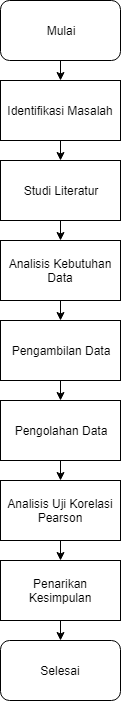
|  |  |
| --- | --- |
| **Prioritas** | **Product Backlog** |
| **3** | **Persetujuan Pengguna dan Petunjuk Penggunaan Aplikasi** |
|  | Penyusunan persetujuan pengguna dan petunjuk penggunaan aplikasi |
|  | Mengembangkan tampilan persetujuan pengguna, petunjuk dan ketentuan survey |
|  | Mengembangkan metode untuk mengecek apakah pengguna bersedia berpartisipasi dalam penelitian atau tidak |
|  | Mengembangkan tampilan pengisian identitas pengguna |
|  | Mengembangkan fungsi menampilkan pesan jika data pengguna belum lengkap |
|  | *Sprint overview* |

Tabel 3.10. *Sprint* 4 : sinkronisasi Firebase, autentikasi, dan penyelesaian akhir

|  |  |
| --- | --- |
| **Prioritas** | **Product Backlog** |
| **4** | **Sinkronisasi Firebase, Autentikasi, dan Penyelesaian Akhir** |
|  | Perancangan basis data Firebase untuk menyimpan data kuesioner dan penggunaan aplikasi |
|  | Mengembangkan metode untuk menyimpan data hasil kuesioner ke dalam Firebase |
|  | Mengembangkan metode untuk menyimpan data durasi dan frekuensi penggunaan aplikasi ke dalam Firebase |
|  | Pengaturan autentikasi pengguna pada Firebase dan mengembangkan metode untuk autentikasi pengguna pada aplikasi |
|  | Menyusun urutan fitur-fitur aplikasi ke dalam alur kerja aplikasi |
|  | *Sprint overview* |

## Prosedur Analisis Uji Korelasi

Analisis uji korelasi dilakukan melalui serangkaian tahapan. Tahapan-tahapan tersebut diilustrasikan melalui diagram alir pada Gambar 3.3. Tiap tahapan tersebut akan dijabarkan secara urut sebagai berikut.



Gambar 3.3. Diagram Alir Prosedur Analisis Uji Korelasi

### Identifikasi Masalah dan Studi Literatur

Penelitian ini merupakan salah satu bagian dari penelitian induk yang dikerjakan dalam tim. Identifikasi masalah merupakan tahap pertama dalam penelitian ini. Kegiatan ini dilakukan dengan melakukan diskusi kelompok secara berkala dengan tim penelitian induk. Kegiatan identifikasi masalah menghasilkan sejumlah kata kunci terkait studi literatur yang perlu dilakukan. Studi literatur dilakukan terhadap kata kunci-kata kunci yang dihasilkan dari proses identifikasi masalah. Kata kunci-kata kunci tersebut antara lain personalisasi pengguna, bentuk-bentuk pembelajaran, gaya belajar, pembelajaran informal, dan idenfikasi gaya belajar. Fokus utama pada studi literatur adalah menemukan penelitian-penelitian kunci yang telah dilakukan untuk dijadikan acuan. Studi literatur pada penelitian-penelitian kunci tersebut memberikan gambaran mengenai metodologi penelitian yang perlu dilakukan. Berdasarkan metodologi yang ditemukan, terdapat sejumlah aspek utama yang perlu ditentukan pada penelitian ini. Aspek-aspek tersebut adalah model gaya belajar dan media pengamatan. Model gaya belajar yang digunakan adalah VARK, sedangkan media pengamatan yang digunakan adalah *smartphone*. Pemahaman terhadap model gaya belajar dan media pengamatan dapat mendasari kegiatan selanjutnya, yaitu analisis kebutuhan data.

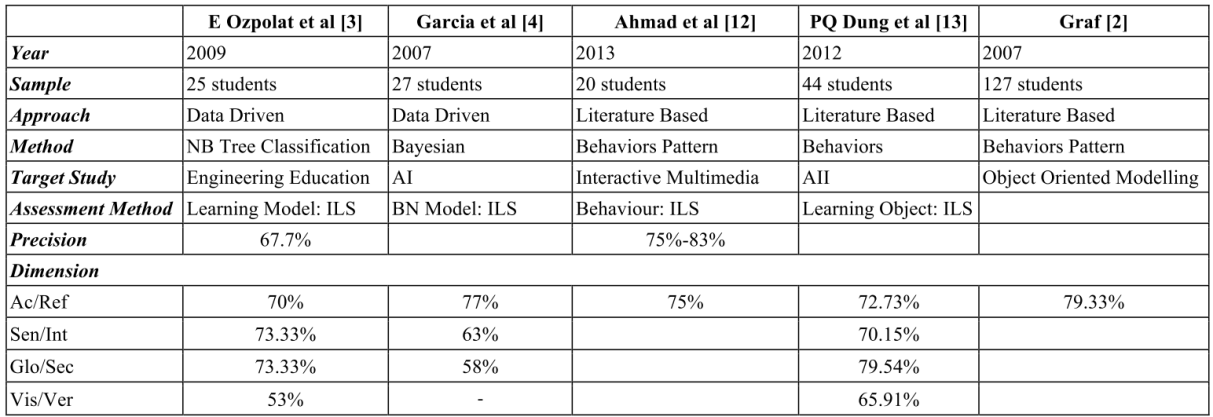
### Analisis Kebutuhan Data

Penelitian ini akan melakukan analisis uji korelasi, sehingga membutuhkan 2 data sebagai peubah yang akan dibandingkan. Dua data tersebut adalah data nilai kuesioner VARK dan data hasil perekaman aktivitas responden pada *smartphone*. Perlu dilakukan analisis kebutuhan data untuk menentukan data apa saja yang dibutuhkan pada perekaman aktivitas responden di *smartphone*. Analisis kebutuhan data tersebut dilakukan berdasarkan model gaya belajar dan media pengamatan yang dipilih. Data yang dibutuhkan merupakan kombinasi dari dua buah aspek, yaitu fitur dan pola perilaku. Fitur merupakan konten-konten yang tersedia pada media pengamatan. Pola perilaku merupakan pola yang terbentuk dari hasil interaksi pengguna dengan fitur-fitur yang tersedia. Kombinasi dari kedua aspek tersebut akan menghasilkan sejumlah data yang dapat dikelompokkan sesuai kelas-kelas yang terdapat pada model gaya belajar. Berdasarkan studi literatur, fitur yang perlu diamati adalah kegunaan aplikasi-aplikasi yang digunakan pada *smartphone*, sedangkan pola perilaku yang perlu diamati adalah frekuensi dan durasi. Sehingga, data yang dibutuhkan adalah frekuensi dan durasi penggunaan aplikasi tertentu pada *smartphone*. Pengelompokan aplikasi-aplikasi ke dalam kelas-kelas VARK akan dibahas lebih lanjut pada tahapan pengolahan data. Setelah mengetahui data apa saja yang dibutuhkan, maka dapat dilakukan pengambilan data.

### Pengambilan Data

Pengumpulan data membutuhkan responden sejumlah minimal 25 orang. Penentuan jumlah responden dilakukan berdasarkan jumlah responden terkecil yang dimiliki oleh penelitian-penelitian sebelumnya. Sumber yang dijadikan acuan merupakan tabel perbandingan dari penelitian yang dilakukan Hasibuan *et. al* seperti yang ditampilkan pada Tabel 3.11. Pengumpulan data nilai kuesioner VARK dilakukan melalui pengisian kuesioner resmi yang terdapat pada situs <http://vark-learn.com/the-vark-questionnaire/>. Sedangkan pengumpulan data penggunaan aplikasi *smartphone* responden dilakukan menggunakan aplikasi piranti bergerak berbasis Android.

Tabel 3.11. Tabel Perbandingan Penelitian-Penelitian Sebelumnya [7]



### Pengolahan Data

Terdapat 3 tahap pengolahan terhadap data yang telah diambil, yaitu pembersihan data, pengelompokan data, dan normalisasi data. Tidak semua aplikasi bersifat relevan bagi penelitian ini. Terdapat sejumlah aplikasi yang tidak dapat dikategorikan ke dalam kelas manapun dari VARK, seperti aplikasi lampu senter, penyetelan *smartphone*, aplikasi untuk ekstraksi berkas, dll. Maka, perlu dilakukan pembersihan data dari aplikasi-aplikasi yang tidak relevan.Data yang telah disaring akan dikelompokkan ke dalam kelas-kelas VARK berdasarkan atribut kegunaan aplikasi. Adapun daftar pengelompokan kegunaan aplikasi ke dalam kelas-kelas VARK dapat dilihat pada Tabel 3.12.

Tabel 3.12. Tabel Perbandingan Penelitian-Penelitian Sebelumnya

|  |  |
| --- | --- |
| Kelas | Kegunaan Aplikasi |
| V | Pemutar video |
| Pembuka gambar |
| Pembuka komik |
| Penampil peta dan navigasi |
| Penampil konten-konten visual lainnya |
| A | Pemutar musik |
| Komunikasi |
| Sosial media |
| R | Pembaca berita |
| Pembaca buku |
| Penulis catatan dan agenda |
| Penampil konten-konten tekstual terstruktur lainnya |
| K | Permainan |
| Pengambil dan penyunting foto atau gambar |
| Pengambil dan penyunting video |
| Desain grafis |
| Aplikasi-aplikasi bersifat praktis lainnya |

Durasi penggunaan *smartphone* secara keseluruhan dari tiap responden cenderung bervariasi. Tingkat penggunaan aplikasi yang rendah bagi seorang responden dapat terbilang tinggi bagi responden lain, jika durasi penggunaan *smartphone* totalnya lebih rendah. Maka, perlu dilakukan normalisasi data untuk menangani isu tersebut. Normalisasi dilakukan melalui perangkat lunak Microsoft Excel dengan menerapkan rumus berikut:

(2)

dimana:

x = nilai asli

x’= nilai ternormalisasi

Tahapan ini akan menghasilkan data berupa nilai ternormalisasi dari durasi dan frekuensi terhadap kelas VARK tertentu.

### Analisis Uji Korelasi Pearson

Analisis uji korelasi dilakukan terpisah untuk tiap pola perilaku, yaitu durasi dan frekuensi akses, terhadap nilai kuesioner VARK. Uji korelasi melibat 2 peubah. Peubah pertama adalah nilai ternormalisasi total dari durasi atau frekuensi akses terhadap kelas VARK tertentu dari seluruh responden. Peubah kedua merupakan nilai kuesioner VARK total dari seluruh responden. Uji korelasi dilakukan bagi tiap kelas V, A, R, dan K untuk tiap data responden. Uji korelasi akan dilakukan dengan dukungan perangkat lunak SPSS.

BAB IV  
HASIL DAN PEMBAHASAN

## *Pengembangan Aplikasi* Usage Statistic Recorder

Bagian pertama dari penelitian ini adalah pengembangan aplikasi berbasis Android untuk mencatat statistik penggunaan aplikasi-aplikasi pada sebuah *smartphone*. Data statistik yang dicatat adalah data frekuensi dan durasi penggunaan aplikasi. Data tersebut akan diteruskan ke basis data Firebase secara *online*. Selain itu, aplikasi ini juga berfungsi untuk melakukan pengisian kuesioner VARK sekaligus menyimpan hasil pengisian kuesioner ke basis data Firebase. Data frekuensi dan durasi penggunaan aplikasi *smartphone* serta data hasil kuesioner VARK akan digunakan dalam analisis uji korelasi pada sub-bab berikutnya. Berdasarkan *product backlog* yang telah dijabarkan pada bab 3, secara garis besar aplikasi ini akan bekerja dengan urutan seperti pada Gambar 4.1.

Gambar 4.1. Urutan kerja aplikasi

Pertama, pengguna akan diarahkan ke halaman petunjuk dan ketentuan survey yang akan dilakukan. Jika pengguna setuju untuk melanjutkan survey, kemudian pengguna akan diarahkan ke halaman untuk mengisi data diri pengguna. Setelah itu, pengguna akan mengisi kuesioner VARK yang terdiri dari 16 pertanyaan dengan 4 pilihan jawaban untuk masing-masing pertanyaan. Setelah survey selesai dilakukan, aplikasi akan menjalankan layanan di balik layar (*background service*) untuk mencatat frekuensi dan durasi penggunaan aplikasi pada *smartphone* pengguna. Terakhir, data durasi dan frekuensi penggunaan aplikasi serta data hasil kuesioner VARK disimpan ke basis data Firebase secara *online*.

Dalam mengembangkan aplikasi ini, metode pengembangan yang digunakan adalah metode SCRUM yang terdiri dari beberapa *sprint*. Tiap *sprint* bertujuan untuk menyelesaikan satu atau lebih fitur aplikasi. *Sprint* dimulai dari fitur dengan skala prioritas tinggi terlebih dahulu. Hasil dari tiap-tiap *sprint* akan dijabarkan lebih lanjut.

### *Sprint 1* : Layanan Pencatat Data Penggunaan Aplikasi

Fitur utama dari aplikasi ini adalah layanan untuk mencatat data penggunaan aplikasi *smartphone*, yaitu durasi dan frekuensi penggunaan tiap aplikasi. Untuk mendapatkan data durasi dan frekuensi penggunaan aplikasi, dibutuhkan sebuah *Application Programmable Interface* (API) bernama *App Usage Statistic*. API ini secara otomatis sudah tersedia pada *smartphone* dengan sistem operasi Android 5.0 (Lollipop) atau versi lebih baru. Sehingga untuk memperoleh data durasi dan frekuensi penggunaan aplikasi cukup dengan cara mengakses data yang telah disediakan oleh API ini.

Langkah pertama yang dilakukan adalah memperbolehkan akses aplikasi terhadap API tersebut. Perizinan akses ini dilakukan dengan menuliskannya ke dalam *file* AndroidManifest.xml yang terdapat dalam folder Android Studio Project. Contoh kode untuk menuliskan perizinan akses terhadap API dapat dilihat pada Gambar 4.2. Karena API ini termasuk ke dalam daftar *permission* yang dilindungi oleh sistem operasi Android, maka perlu dituliskan pula perintah untuk mengabaikan (*ignore*) kondisi tersebut.



Gambar 4.2. Contoh kode untuk meberikan izin akses API

Perizinan untuk mengakses API ini berbeda dengan penggunaan *permission* di Android pada umumnya. Pada *permission* lain, setelah menuliskannya pada *file* AndroidManifest.xml maka fitur yang bersangkutan langsung dapat digunakan / diakses. Namun untuk mengakses API ini, selain menuliskannya pada *file* AndroidManifest.xml, pengguna juga harus secara aktif membuka menu pengaturan pada perangkat dan memperbolehkan aplikasi untuk mengakses *App Usage Statistic*. Jika hal ini tidak dilakukan, maka aplikasi tetap dapat berjalan namun tidak akan bisa mengakses *App Usage Statistic*.

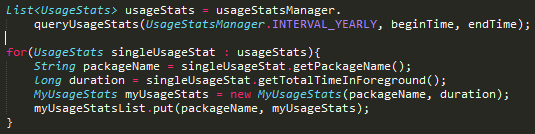
Untuk memberikan kemudahan kepada pengguna, maka diberikan menu akses pintas (*shortcut*) untuk membuka menu pengaturan tersebut seperti pada gambar 4.3 a). Tombol pertama berfungsi untuk membuka menu pengaturan perangkat, sedangkan tombol kedua berfungsi untuk menuju ke halaman / menu berikutnya. Jika tombol pertama ditekan, maka akan muncul menu pengaturan akses *App Usage Statistic* seperti pada gambar 4.3 b).

|  |  |
| --- | --- |
| E:\Documents\Semester 8\SKRIPSI BARU\Diagram Gambar dll\Screenshot_2018-06-26-14-23-36_com.example.meizar.learningstyleapp.png  a) | E:\Documents\Semester 8\SKRIPSI BARU\Diagram Gambar dll\Screenshot_2018-06-26-14-23-47_com.android.settings.png  b) |

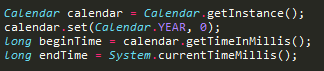
Gambar 4.3. Menu akses pintas untuk membuka pengaturan perangkat : a) Tampilan antarmuka pada aplikasi; b) Tampilan menu pengaturan izin penggunaan API

Setelah aplikasi diperbolehkan untuk mengakses API melalui langkah-langkah tersebut, maka data durasi dan frekuensi penggunaan aplikasi *smartphone* dapat diperoleh. Namun untuk memperoleh dua data ini dilakukan dengan cara yang berebeda. Untuk memperoleh data durasi, dilakukan *query* terhadap UsageStats seperti pada Gambar 4.4. *Query* dilakukan dengan memasukkan tiga buah parameter, yaitu jenis interval (Daily, Weekly, Monthly, Yearly, atau Best), waktu mulai (beginTime), dan waktu selesai (endTime). Interval yang dipilih adalah interval Yearly karena interval ini memberikan data historis paling lengkap jika dibandingkan interval lainnya. Pengaturan waktu mulai dan waktu selesai dilakukan seperti pada potongan kode pada Gambar 4.5.

Untuk parameter waktu mulai diatur sedini mungkin, yaitu tahun ke-0 atau tahun 1970 berdasarkan ketetapan standar waktu Android. Sedangkan untuk parameter waktu selesai diatur seakhir mungkin, yaitu waktu sekarang (ketika baris program Sytem.currentTimeMillis() dieksekusi). Dengan cara ini, maka akan diperoleh seluruh data statistik yang tersimpan di dalam API. Hasil dari *query* ini adalah sebuah daftar (*list*) berisi Usage Stats perangkat tersebut. Nama aplikasi diperoleh melalui fungsi getPackageName() sedangkan durasi penggunaan aplikasi diperoleh melalui getTotalTimeInForeGround().

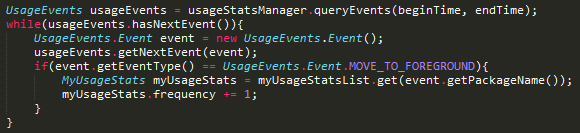


Gambar 4.4. Contoh kode program untuk melakukan *query* terhadap UsageStats



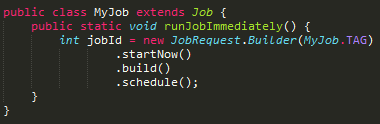
Gambar 4.5. Contoh kode program untuk mengatur beginTime dan endTime

Kemudian untuk mendapatkan data frekuensi akses terhadap suatu aplikasi, *query* dilakukan terhadap obyek yang berbeda, yaitu UsageEvents seperti pada Gambar 4.6. Untuk mendapatkan data UsageEvents, parameter yang dibutuhkan hanya dua buah, yaitu beginTime dan endTime. Sama seperti saat mengakses data UsageStats, parameter beginTime diatur mulai tahun ke-0 dan parameter endTime diatur waktu sekarang seperti pada Gambar 4.5 sebelumnya. Hasil *query* ini akan menghasilkan daftar (*list*) kejadian-kejadian (*events*) selama interval beginTime dan endTime. Untuk mendapatkan data frekuensi akses suatu aplikasi, maka dilakukan *filtering* terhadap kejadian yang ada, yaitu hanya kejadian dengan jenis MOVE\_TO\_FOREGROUNDsaja yang akan dihitung. Untuk tiap aplikasi, akan diakumulasikan berapa kali aplikasi tersebut berpindah dari *background* menuju *foreground*, yang menandakan bahwa aplikasi tersebut dibuka / diakses oleh pengguna.



Gambar 4.6. Contoh kode program untuk melakukan *query* terhadap UsageEvents

Karena fungsi untuk mendapatkan data penggunaan aplikasi ini bekerja di balik layar, maka fungsi ini perlu dideklarasikan sebagai sebuah layanan (*service*). Digunakan *library* bantuan untuk mengembangkan *service* ini, yaitu Evernote Android Job. Sebuah *Job* bernama MyJob dibuat untuk menampung fitur pembacaan data frekuensi dan durasi ini seperti pada Gambar 4.10. Untuk memulai *Job* ini, cukup memanggil fungsi runJobImmediately() pada program utama.



Gambar 4.7. Kelas MyJob berjenis Android Job

Untuk menguji hasil akhir *Sprint* 1, dilakukan pencatatan durasi dan frekuensi penggunaan aplikasi dan menampilkannya dalam *log* aplikasi. Diperoleh hasil seperti pada Gambar 4.8. Tampak bahwa aplikasi dapat merkeam durasi dan frekuensi penggunaan aplikasi lain. Data durasi diperoleh dalam bentuk satuan milisekon. Beberapa aplikasi akan memberikan data frekuensi dengan nilai 0 meskipun data durasinya tidak 0. Hal ini dikarenakan perbedaan cara mengakses data durasi dan frekuensi seperti yang telah dijelaskan sebelumnya. Data durasi yang diperoleh melalui *query* UsageStats akan memberikan data historis yang lebih lengkap (lebih lama) jika dibandingkan dengan data frekuensi yang diperoleh melalui *query* UsageEvents.

|  |
| --- |
| com.android.captiveportallogin  duration: 119779  frequency: 0  com.android.chrome  duration: 4826552  frequency: 35  com.facebook.appmanager  duration: 378780  frequency: 0  jp.naver.line.android  duration: 24123634  frequency: 2285 |

Gambar 4.8. Contoh *log* data durasi dan frekuensi penggunaan aplikasi

Dengan dikembangkannya fungsi-fungsi tersebut, maka dapat dikatakan *Sprint 1* telah selesai dilakukan. Aplikasi sudah memiliki akses pintas ke menu pengaturan perangkat pada bagian *App usage Access*. Aplikasi juga dapat menangkap data durasi dan frekuensi penggunaan aplikasi-aplikasi lain. Selain itu, fungsi untuk menangkap data durasi dan frekuensi dapat berjalan di *background process*. Daftar fungsi yang telah dibuat sesuai dengan *Product Backlog* yang telah disusun sebelumnya, seperti pada Tabel 4.1.

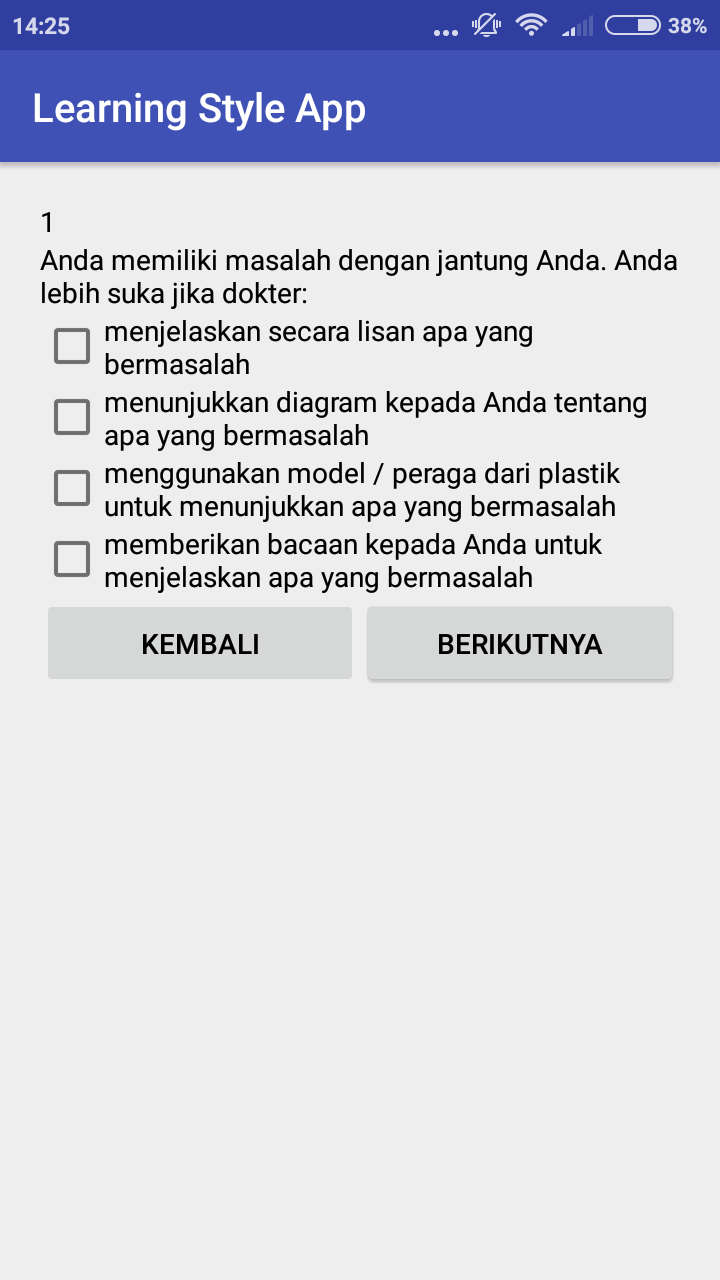
Tabel 4.1. *Sprint* 1 *Overview*

|  |  |  |
| --- | --- | --- |
| **Prioritas** | **Product Backlog** | **Keterangan** |
| **1** | **Layanan Penangkap Data Penggunaan Aplikasi** |  |
|  | Membuat menu akses pintas ke pengaturan perangkat bagian *App Usage Access* | Selesai |
|  | Mengembangkan layanan penangkap data durasi penggunaan aplikasi | Selesai |
|  | Mengembangkan layanan penangkap data frekuensi penggunaan aplikasi | Selesai |
|  | Membuat layanan berjalan pada *background* | Selesai |

### *Sprint 2* : Fitur Pengisian Kuesioner VARK

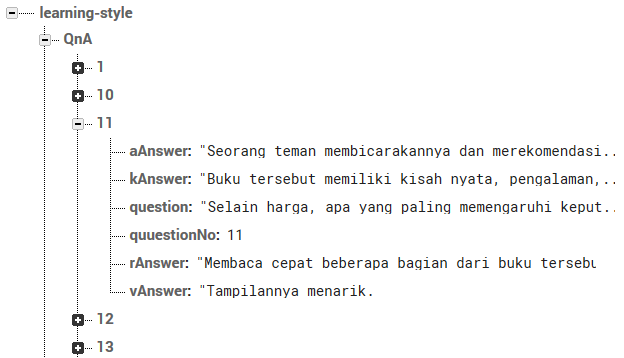
Pada sprint kedua, fitur yang akan diselesaikan adalah fitur pengisian kuesioner VARK. Fitur ini diperlukan untuk memudahkan pengumpulan data responden. Tanpa fitur ini, maka responden harus mengisi kuesioner VARK secara manual, baik itu melalui kertas maupun formulir *online*. Lalu setelah itu responden masih harus memasang aplikasi yang telah dikembangkan agar data durasi dan frekuensi dapat diperoleh. Dengan adanya fitur ini, maka responden dapat langsung mengisi kuesioner melalui aplikasi.

Tampilan halaman pengisian kuesioner VARK dapat dilihat seperti pada Gambar 4.9. Bagian atas halaman berisi pertanyaan dan nomor pertanyaan. Tersedia empat pilihan jawaban dalam bentuk *checkbox* sehingga responden dapat memilih lebih dari satu pilihan jawaban atau tidak memilih sama sekali. Pada bagian bawah halaman terdapat dua buah tombol navigasi, yaitu tombol untuk kembali ke pertanyaan sebelumnya dan tombol untuk menuju ke pertanyaan berikutnya.



Gambar 4.9. Tampilan antarmuka pengisian kuesioner VARK

Daftar pertanyaan dan jawaban survey disimpan di basis data Firebase pada node QnA seperti pada Gambar 4.10. Setiap obyek QnA memiliki enam buah pasangan *key* dan *value*, yaitu pertanyaan (*question*), nomor pertanyaan (*questionNo*), dan masing-masing satu pilihan jawaban untuk tiap dimensi VARK (*vAnswer*, *aAnswer*, *rAnswer*, dan k*Answer*). Agar obyek QnA tersebut dapat ditampilkan dalam aplikasi, maka perlu dibuat POJO terkait dengan nama QuestionAndAnswers seperti pada Gambar 4.11. Selain memiliki atribut-atribut tersebut, POJO ini memiliki atribut tambahan berupa *vPoint, aPoint, rPoint, dan kPoint*. Atribut-atribut tambahan ini berfungsi untuk menyimpan data poin VARK dari suatu pertanyaan yang nantinya akan diakumulasikan di akhir survey.



Gambar 4.10. Sumber data pertanyaan dan jawaban survey VARK yang tersimpan di Firebase



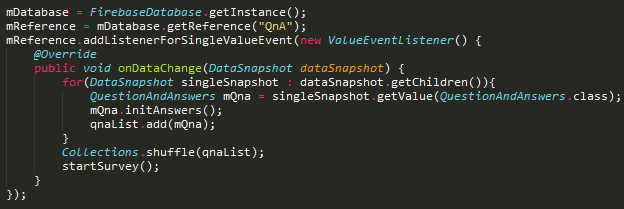
Gambar 4.11. POJO *class* QuestionAndAnswers

Dalam survey VARK, pilihan jawaban yang akan ditampilkan dalam kondisi urutan yang acak. Artinya, pilihan jawaban preferensi *visual* tidak selalu berada pada pilihan jawaban pertama. Bisa jadi pilihan jawaban preferensi *visual* berada pada urutan kedua, ketiga, atau keempat. Begitu juga dengan preferensi pilihan jawaban lainnya, yaitu *aural*, *read/write*, dan *kinesthetic*. Contoh kode program pada Gambar 4.12 akan mengacak urutan jawaban survey. Pemanggilan fungsi Collections.shuffle() akan mengacak urutan jawaban survey.



Gambar 4.12. Contoh kode program untuk mengacak urutan pilihan jawaban survey

Setelah POJO dibuat, maka data dari Firebase dapat diambil dan ditampilkan ke dalam aplikasi. Contoh kode untuk mengambil data pertanyaan dan jawaban dapat dilihat pada Gambar 4.13. Selain urutan pilihan jawaban yang teracak, urutan pertanyaan yang ditampilkan juga dalam kondisi acak. Untuk itulah perintah Collections.shuffle() kembali dipanggil untuk mengacak urutan pertanyaan. Hasilnya, urutan pertanyaan yang muncul akan berbeda setiap kali aplikasi dijalankan, seperti pada Gambar 4.14.

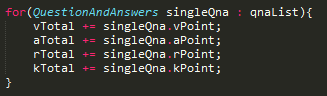


Gambar 4.13. Contoh kode program untuk mengambil data pertanyaan dan jawaban dari Firebase

|  |  |
| --- | --- |
| E:\Documents\Semester 8\SKRIPSI BARU\Diagram Gambar dll\Screenshot_2018-06-26-14-25-52_com.example.meizar.learningstyleapp.png  a) | E:\Documents\Semester 8\SKRIPSI BARU\Diagram Gambar dll\Screenshot_2018-06-26-20-59-37_com.example.meizar.learningstyleapp.png  b) |

Gambar 4.14. Hasil pengacakan urutan pertanyaan dan jawaban survey: a) Aplikasi dijalankan pertama kali; b) Aplikasi dijalankan kedua kali

Sesudah 16 buah pertanyaan dijawab, aplikasi akan menghitung hasil poin untuk masing-masing dimensi VARK. Poin tersebut diperoleh dari akumulasi poin VARK tiap pertanyaan dengan cara seperti pada Gambar 4.15. Untuk mengecek hasil akhir dari survey tersebut, hasil akan ditampilkan sementara pada *log* data seperti pada Gambar 4.16. Diperoleh bahwa aplikasi telah dapat digunakan untuk melakukan pengisian survey kuesioner VARK.

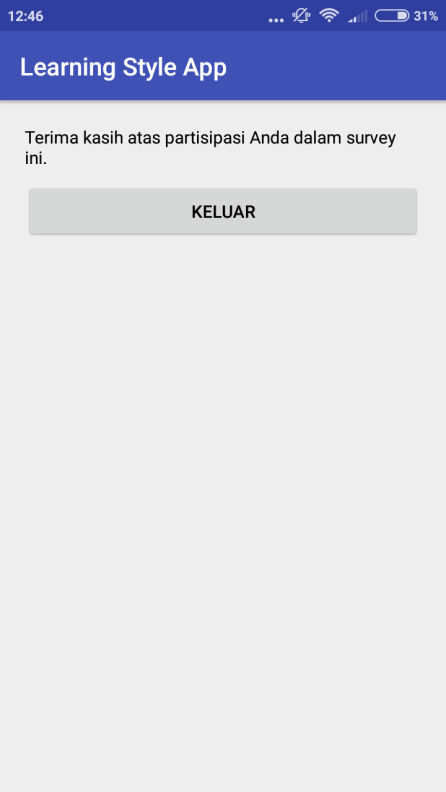


Gambar 4.15. Contoh kode program untuk menghitung hasil survey

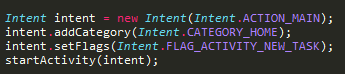
|  |
| --- |
| visual : 9  aural : 7  read/write : 10  kinesthetic : 11 |

Gambar 4.16. Contoh *log* data hasil survey VARK

Halaman akhir akan ditampilkan setelah pengguna menyelesaikan survey seperti pada Gambar 4.17. Halaman tersebut berisi pesan terima kasih dan satu buah tombol untuk keluar dari aplikasi. Untuk keluar dari aplikasi, Android tidak menyediakan fungsi atau perintah untuk menutup aplikasi secara total. Sehingga cara untuk keluar dari aplikasi adalah dengan mengarahkan aplikasi ke tampilan beranda (*home screen*) dengan kode program seperti pada Gambar 4.18. Selain tombol keluar, kode program tersebut juga diimplementasikan pada tombol kembali (*back*) yang terdapat pada tiap perangkat. Sehingga ketika tombol kembali ditekan aplikasi akan tertutup, bukan kembali ke halaman survey.



Gambar 4.17. Tampilan halaman akhir



Gambar 4.18. Contoh kode program untuk keluar dari aplikasi

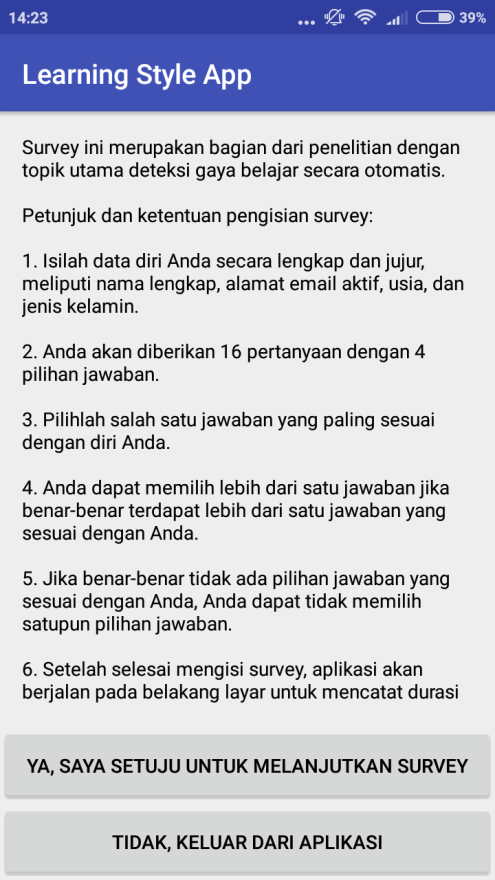
Setelah menyelesaikan fungsi-fungsi tersebut, maka *Sprint* 2 telah selesai dilaksanakan. Daftar pertanyaan dan jawaban suervey VARK sudah tersimpan di dalam Firebase. Aplikasi sudah dapat mengambil data pertanyaan dan jawaban dari Firebase. Kemudian, aplikasi juga dapat menampilkan pertanyaan dan jawaban secara acak. Terakhir, aplikasi juga dapat menghitung akumulasi poin VARK sebagai hasil akhir dari survey. Daftar fungsi yang telah dikembangkan tersebut sesuai dengan *Product Backlog* yang telah disusun sebelumnya seperti pada Tabel 4.2.

Tabel 4.2. *Sprint* 2 *Overview*

|  |  |  |
| --- | --- | --- |
| **Prioritas** | **Product Backlog** | **Keterangan** |
| **2** | **Kuesioner VARK** |  |
|  | Mempersiapkan daftar pertanyaan dan jawaban kuesioner serta menterjemahkannya ke dalam bahasa Indonesia | Selesai |
|  | Memasukkan daftar pertanyaan dan jawaban kuesioner ke dalam Firebase | Selesai |
|  | Mengembangkan tampilan Android pengisian kuesioner VARK | Selesai |
|  | Mengembangkan metode untuk mengacak urutan pertanyaan kuesioner | Selesai |
|  | Mengembangkan metode untuk mengacak urutan jawaban tiap pertanyaan kuesioner | Selesai |
|  | Mengembangkan metode untuk menampilkan pertanyaan dan kuesioner | Selesai |
|  | Mengembangkan metode untuk menghitung hasil kuesioner | Selesai |

### *Sprint 3* : Persetujuan Pengguna, Petunjuk Penggunaan Aplikasi, dan Pengisian Identitas Responden

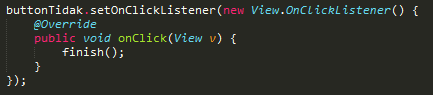
Sebelum survey dilaksanakan, diperlukan persetujuan dari pengguna / responden. Pengguna berhak mengetahui ketentuan survey secara lengkap beserta data apa saja yang akan diambil dari survey ini. Untuk itulah fitur ini dikembangkan. Tampilan halaman persetujuan pengguna dan petunjuk penggunaan aplikasi dapat dilihat pada Gambar 4.19. Terdapat sembilan poin terkait petunjuk dan ketentuan mengenai survey yang akan dilaksanakan seperti pada Tabel 4.3. Apabila pengguna telah memahami poin-poin tersebut dan bersedia untuk berpartisipasi, maka pengguna dapat menekan tombol setuju dan akan diarahkan ke halaman berikutnya. Namun apabila pengguna tidak bersedia untuk melanjutkan survey, maka pengguna dapat menekan tombol tidak dan aplikasi akan tertutup secara otomatis. Kode instruksi program untuk keluar dari aplikasi cukup sederhana, yaitu perintah finish() seperti pada Gambar 4.20.



Gambar 4.19. Tampilan persetujuan pengguna dan petunjuk penggunaan aplikasi

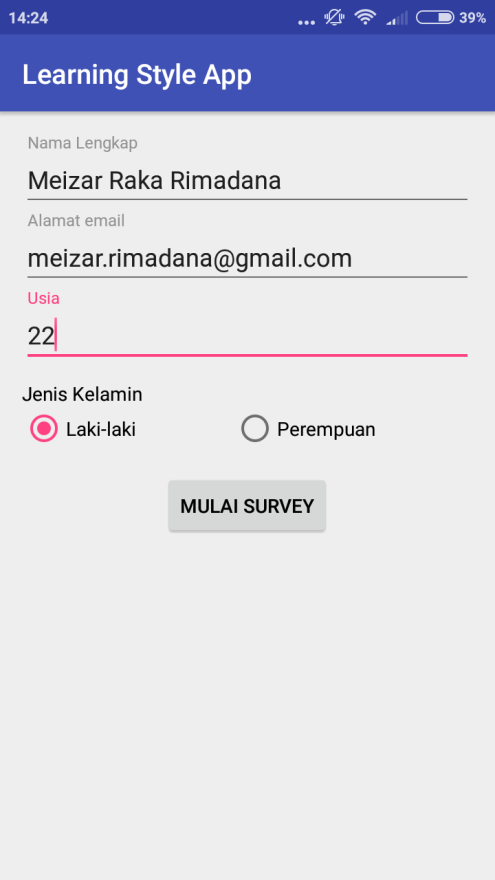
Tabel 4.3. Daftar petunjuk dan ketentuan survey

|  |  |
| --- | --- |
| **Nomor** | **Petunjuk dan Ketentuan Survey** |
| 1. | Isilah data diri Anda secara lengkap dan jujur, meliputi nama lengkap, alamat email aktif, usia, dan jenis kelamin. |
| 2. | Anda akan diberikan 16 pertanyaan dengan 4 pilihan jawaban. |
| 3. | Pilihlah salah satu jawaban yang paling sesuai dengan diri Anda. |
| 4. | Anda dapat memilih lebih dari satu jawaban jika benar-benar terdapat lebih dari satu jawaban yang sesuai dengan Anda. |
| 5. | Jika benar-benar tidak ada pilihan jawaban yang sesuai dengan Anda, Anda dapat tidak memilih satupun pilihan jawaban. |
| 6. | Setelah selesai mengisi survey, aplikasi akan berjalan pada belakang layar untuk mencatat durasi dan frekuensi penggunaan aplikasi-aplikasi smartphone Anda. |
| 7. | Aplikasi hanya akan mencatat data durasi dan frekuensi penggunaan aplikasi-aplikasi smartphone Anda. Aplikasi ini tidak akan mencatat data lainnya. |
| 8. | Data hasil survey ini terjamin kerahasiannya dan hanya akan digunakan dalam kepentingan penelitian ini. |
| 9. | Jika di kemudian hari terdapat indikasi penyalahgunaan data milik Anda, Anda berhak untuk melakukan tuntutan sesuai dengan hukum yang berlaku. |



Gambar 4.20. Contoh kode program untuk keluar dari aplikasi

Berikutnya, pengguna akan diarahkan ke halaman pengisian identitas seperti pada Gambar 4.21. Data diri yang perlu diisi oleh responden adalah nama lengkap, alamat email aktif, usia, dan jenis kelamin. Nama diperlukan untuk membedakan antara responden satu dengan responden yang lain. Data email diperlukan untuk kebutuhan autentikasi pengguna. Sedangkan data tambahan berupa usia dan jenis kelamin apabila dibutuhkan dapat digunakan untuk analisis lebih lanjut pada penelitian berikutnya, misalnya untuk mengetahui perbedaan karakteristik gaya belajar berdasarkan jenis kelamin. Keempat data tersebut harus diisi oleh pengguna. Jika terdapat satu atau lebih data yang belum diisi, maka akan muncul pesan pemberitahuan pada aplikasi seperti pada Gambar 4.22.



Gambar 4.21. Halaman pengisian identitas responden

|  |  |
| --- | --- |
| E:\Documents\Semester 8\SKRIPSI BARU\Diagram Gambar dll\Screenshot_2018-06-26-14-24-28_com.example.meizar.learningstyleapp.png  a)  E:\Documents\Semester 8\SKRIPSI BARU\Diagram Gambar dll\Screenshot_2018-06-26-14-25-00_com.example.meizar.learningstyleapp.png  c) | E:\Documents\Semester 8\SKRIPSI BARU\Diagram Gambar dll\Screenshot_2018-06-26-14-24-47_com.example.meizar.learningstyleapp.png  b)  E:\Documents\Semester 8\SKRIPSI BARU\Diagram Gambar dll\Screenshot_2018-06-26-14-25-26_com.example.meizar.learningstyleapp.png  d) |

Gambar 4.22. Pesan yang muncul jika salah satu data belum diisi: a) data nama; b) data alamat email; c) data usia; d) data jenis kelamin

Dengan dikembangkannya fitur-fitur tersebut, maka *Sprint* 3 selesai dilaksanakan. Poin persetujuan pengguna dan petunjuk serta ketentuan survey telah disusun dan ditampilkan ke dalam aplikasi. Aplikasi juga dapat memberikan pilihan kepada pengguna apakah pengguna ingin melanjutkan survey atau tidak. Selain itu, aplikasi juga telah mampu mencatat data diri pengguna serta menampilkan pesan jika terdapat data pengguna yang belum diisi. Fitur-fitur yang telah dikembangkan tersebut sesuai dengan *product backlog* yang telah disusun seperti pada Tabel 4.4.

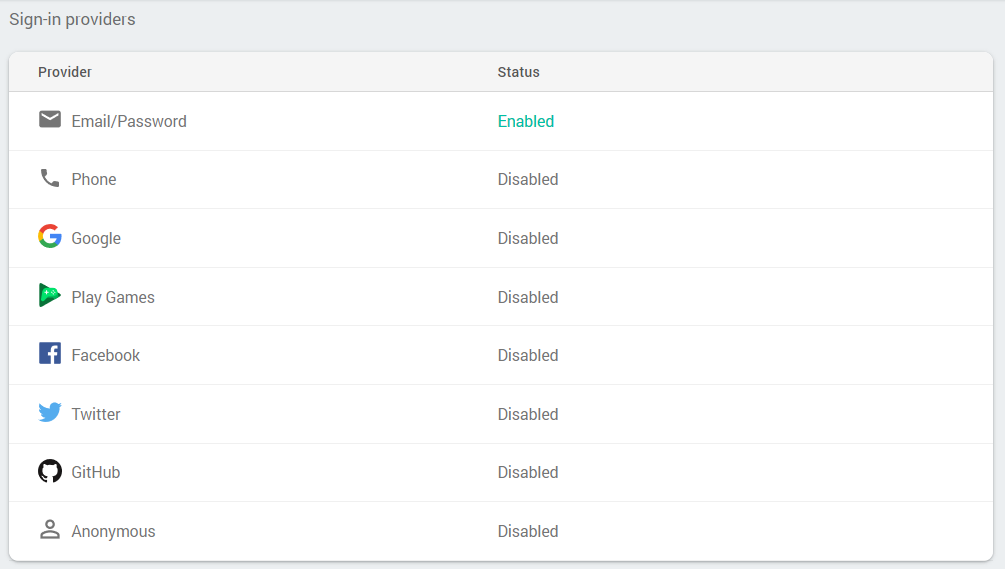
Tabel 4.4. *Sprint* 3 *Overview*

|  |  |  |
| --- | --- | --- |
| **Prioritas** | **Product Backlog** | **Keterangan** |
| **3** | **Persetujuan Pengguna dan Petunjuk Penggunaan Aplikasi** |  |
|  | Penyusunan persetujuan pengguna dan petunjuk penggunaan aplikasi | Selesai |
|  | Mengembangkan tampilan persetujuan pengguna, petunjuk dan ketentuan survey | Selesai |
|  | Mengembangkan metode untuk mengecek apakah pengguna bersedia berpartisipasi dalam penelitian atau tidak | Selesai |
|  | Mengembangkan tampilan pengisian identitas pengguna | Selesai |
|  | Mengembangkan fungsi menampilkan pesan jika data pengguna belum lengkap | Selesai |

### *Sprint 4* : Sinkronisasi Data dengan Firebase

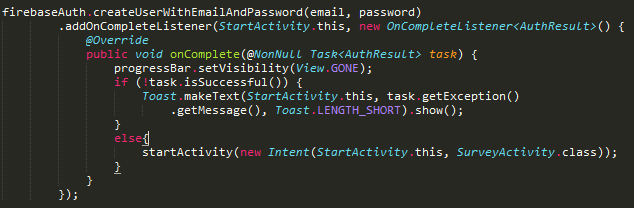
*Sprint* ke-4 bertujuan untuk menghubungkan lebih lanjut antara aplikasi dengan *server* Firebase. Fungsi pertama yang dikembangkan adalah autentikasi pengguna. Hal ini bertujuan untuk membedakan antara satu pengguna dengan pengguna lain. Selain itu, hal ini juga bertujuan untuk membatasi agar seorang pengguna hanya dapat mengisi survey satu kali saja.

Agar aplikasi dapat melakukan proses autentikasi pengguna, maka pada *dashboard* Firebase bagian menu *Sing-in providers* pilihan *Email/Password* diaktifkan seperti pada Gambar 4.23. Sedangkan pilihan autentikasi lain tetap dibiarkan non-aktif. Sehingga, pengguna hanya dapat melakukan autentikasi menggunakan email dan password saja.



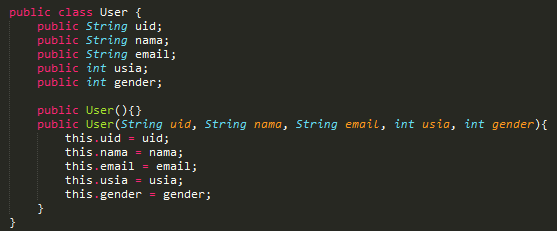
Gambar 4.23. Pengaturan autentikasi pengguna pada *dashboard* Firebase

Proses untuk autentikasi pengguna yang berlangsung pada aplikasi berbeda dengan proses autentikasi pada aplikasi lain. Jika pada aplikasi lain, sebelum melakukan proses *sign-in*, pengguna harus melakukan proses registrasi terlebih dahulu. Hal ini kurang efektif jika diterapkan pada aplikasi ini mengingat sifat aplikasi survey ini adalah aplikasi jangka pendek, bukan aplikasi yang digunakan terus menerus secara harian. Sehingga proses autentikasi pada aplikasi ini diringkas menjadi satu kali proses autentikasi, yaitu proses registrasi dan *sign-in* secara bersamaan. Alamat email yang digunakan untuk autentikasi diambil dari data diri pengguna, sedangkan *password* dibuat sama untuk semua pengguna. Jadi ketika pengguna selesai mengisikan data diri, sekaligus saat itu juga proses autentikasi berlangsung. Contoh kode program untuk melakukan proses autentikasi ini dapat dilihat pada Gambar 4.24. Dua buah kondisi berbeda akan terjadi setelah proses autentikasi berlangsung, yaitu gagal atau berhasil. Jika proses autentikasi gagal, aplikasi akan menampilkan pesan yang menyebabkan autentikasi gagal, misalnya alamat email sudah pernah dipakai sebelumnya. Sebaliknya jika proses autentikasi berhasil, aplikasi akan berpindah ke halaman berikutnya, yaitu halaman pengisian survey.



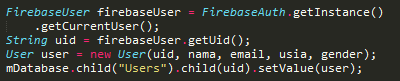
Gambar 4.24. Contoh kode program untuk melakukan proses autentikasi pengguna

Proses menghubungkan aplikasi dengan basis data Firebase yang berikutnya adalah proses penyimpanan data. Data pertama yang akan disimpan adalah data pengguna. Secara *default*, Firebase sebenarnya sudah menyimpan data pengguna berupa alamat email dan *password* yang digunakan untuk proses autentikasi. Namun karena survey ini mengandung data tambahan berupa usia dan jenis kelamin, maka perlu dibuat mekanisme tersendiri untuk menyimpan data pengguna. Agar data pengguna dapat disimpan ke dalam Firebase, data tersebut perlu diubah terlebih dahulu ke dalam sebuahkelas berjenis *Plain Old Java Object* (POJO). Kelas tersebut diberi nama User, sesuai dengan obyek yang akan disimpan seperti pada Gambar 4.25. Kelas User terdiri dari lima buah atribut, yaitu identitas unik pengguna (*uid*), nama, email, usia, dan gender / jenis kelamin.



Gambar 4.25. POJO kelas User

Kemudian obyek User akan disimpan ke basis data Firebase dengan node “Users” berdasarkan *uid* dari pengguna yang bersangkutan seperti pada Gambar 4.26. Firebase secara otomatis akan memberikan *uid* yang berbeda untuk setiap pengguna. Untuk mendapatkan data *uid* tersebut, firebase menyediakan fungsi getUid(). Setelah *uid* diperoleh, obyek User akan dibuat dan dilengkapi dengan data nama, email, usia, dan gender untuk kemudian disimpan ke dalam basis data Firebase. Hasil penyimpanan pada basis data Firebase dapat dilihat pada Gambar 4.27.

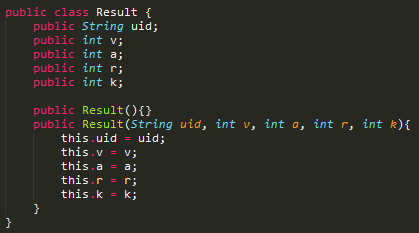


Gambar 4.26. Contoh kode program untuk menyimpan data pengguna



Gambar 4.27. Contoh hasil penyimpanan data pengguna

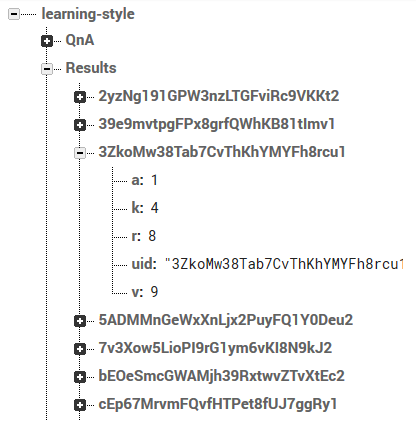
Data berikutnya yang akan disimpan ke dalam basis data Firebase adalah data hasil survey pengguna. Sama seperti proses penyimpanan data penggguna, pertama dibuat sebuah kelas POJO bernama Result seperti pada Gambar 4.28. Kelas Result memiliki lima buah atribut, yaitu identitas unik pengguna (*uid*) dan empat buah atribut yang masing-masing mewakili poin *visual* (v), *aural* (a), *read/write* (r), dan *kinesthetic* (k). Kemudian data tersebut akan disimpan ke dalam basis data Firebase pada node “Result” sesuai dengan *uid* dari masing-masing pengguna seperti pada Gambar 4.29. Hasil dari penyimpanan data survey pengguna ini dapat dilihat pada Gambar 4.30.



Gambar 4.28. Kelas POJO Result

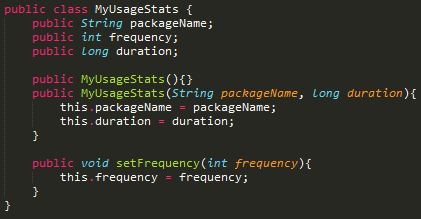


Gambar 4.29. Contoh kode program untuk menyimpan data hasil survey



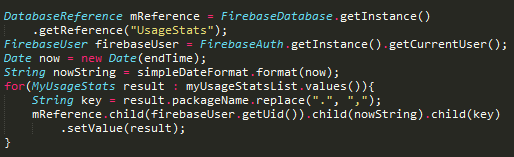
Gambar 4.30. Contoh hasil penyimpanan data survey ke basis data Firebase

Data terakhir yang akan disimpan adalah data penggunaan aplikasi perangkat pengguna. Sama seperti sebelumnya, proses penyimpanan data dimulai dengan pembuatan kelas POJO terlebih dahulu. Kelas yang dibuat diberi nama MyUsageStats seperti pada Gambar 4.31. Kelas MyUsageStats hanya berisi tiga buah atribut, yaitu nama aplikasi (*packageName*), durasi (*duration*), dan frekuensi (*frequency*).

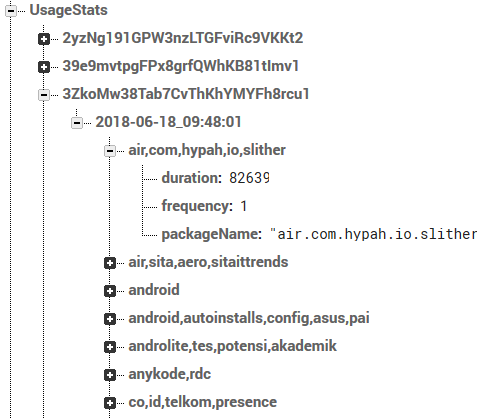


Gambar 4.31. POJO kelas MyUsageStats

Berikutnya, obyek tersebut akan disimpan ke dalam Firebase dalam node UsageStats berdasarkan *Unique ID* (UID) pengguna tersebut seperti pada Gambar 4.32. Data dilengkapi dengan penanda waktu (*timestamp*) terlebih dahulu untuk mengetahui kapan data tersebut diambil. Hal ini bertujuan untuk mengantisipasi apabila terjadi dua kali atau lebih pengiriman data dari pengguna yang sama. Selain itu, nama aplikasi perlu diubah terlebih dahulu yaitu dengan mengganti karakter titik menjadi karakter koma. Hal ini dikarenakan keterbatasan Firebase yang tidak dapat menyimpan obyek dengan nama yang mengandung karakter khusus (“.”, “$”, “[“, “]”, “#”, dan “/”). Contoh hasil penyimpanan data ke dalam Firebase dapat dilihat pada Gambar 4.33.

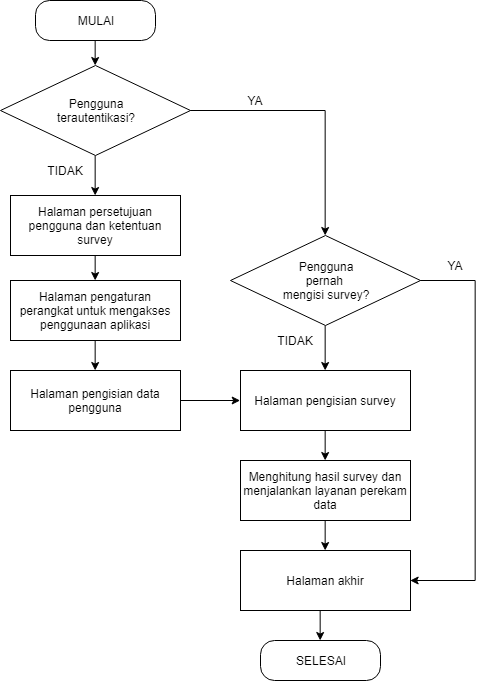


Gambar 4.32. Contoh kode program untuk menyimpan data ke Firebase

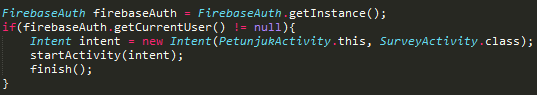


Gambar 4.33. Contoh hasil penyimpanan data ke Firebase

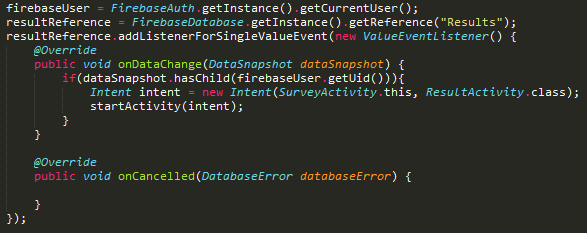
Tahap terakhir dari *Sprint* ke-4 ini adalah menyusun komponen-komponen aplikasi menjadi sebuah aplikasi utuh dan berjalan sesuai dengan alur aplikasi. Alur berjalannya aplikasi dapat dilihat pada Gambar 4.34. Pertama, aplikasi akan memeriksa apakah pengguna telah terautentikasi atau belum. pengguna yang telah terautentikasi berarti pengguna tersebut sudah memasukkan data diri pada aplikasi. Jika pengguna sudah terautentikasi, maka pengguna akan diarahkan langsung ke halaman survey. Namun jika ternyata pengguna juga sudah pernah mengisi survey, maka pengguna akan diarahkan ke halaman akhir. Metode untuk memeriksa apakah pengguna sudah terautentikasi dapat dilihat pada potongan kode seperti pada Gambar 4.35, yaitu dengan cara memeriksa apakah fungsi getCurrentuser() memberikan hasil kosong (*null*) atau tidak. Sedangkan metode untuk memeriksa apakah pengguna sudah mengisi survey atau belum dapat dilihat pada potongan kode seperti pada Gambar 4.36, yaitu dengan cara memeriksa pada basis data Firebase pada node “Results”. Jika pada node tersebut ditemukan data dengan *uid* yang sama dengan *uid* pengguna aplikasi saat ini, berarti pengguna tersebut sudah pernah melakukan pengisian survey.



Gambar 4.34. Diagram alur kerja aplikasi



Gambar 4.35. Contoh kode program untuk memeriksa apakah pengguna sudah terautentikasi atau belum



Gambar 4.36. Contoh kode program untuk memeriksa apakah pengguna sudah pernh mengisi survey atau belum

Pengguna yang belum terautentikasi akan diarahkan pada halaman pengaturan perangkat untuk memperbolehkan aplikasi mengakses data *App Usage Statistics*. Setelah itu, pengguna akan diarahkan ke halaman pengisian data diri pengguna. Kemudian, aplikasi akan menampilkan halaman pengisian survey. Setelah semua pertanyaan selesai dijawab oleh pengguna, aplikasi akan melakukan perhitungan hasil sruvey dan mulai menjalankan layanan untuk mengambil data *App Usage Statistics*. Terakhir, pengguna akan diarahkan ke halaman akhir dan keluar dari aplikasi.

Dengan selesainya penyusunan alur kerja aplikasi tersebut, maka *Sprint* ke-4 telah selesai dilaksanakan. Metode untuk melakukan autentikasi pengguna telah dikembangkan. Data pengguna, hasil survey, dan data durasi serta frekuensi pengaksesan aplikasi telah berhasil tersimpan ke basis data Firebase. Selain itu, fitur-fitur aplikasi telah disusun berdasarkan urutan alur kerja aplikasi. Hal-hal yang dilakukan tersebut sesuai dengan *product backlog* yang telah disusun sebelumnya seperti pada Tabel 4.5.

Tabel 4.5. *Sprint* 4 *Overview*

|  |  |  |
| --- | --- | --- |
| **Prioritas** | **Product Backlog** | **Keterangan** |
| **4** | **Sinkronisasi Firebase, Autentikasi, dan Penyelesaian Akhir** |  |
|  | Perancangan basis data Firebase untuk menyimpan data kuesioner dan penggunaan aplikasi | Selesai |
|  | Mengembangkan metode untuk menyimpan data hasil kuesioner ke dalam Firebase | Selesai |
|  | Mengembangkan metode untuk menyimpan data durasi dan frekuensi penggunaan aplikasi ke dalam Firebase | Selesai |
|  | Pengaturan autentikasi pengguna pada Firebase dan mengembangkan metode untuk autentikasi pengguna pada aplikasi | Selesai |
|  | Menyusun urutan fitur-fitur aplikasi ke dalam alur kerja aplikasi | Selesai |

DAFTAR PUSTAKA

|  |  |
| --- | --- |
| [1] | J. Yang, Z. X. Huang, Y. X. Gao dan H. T. Liu, “Dynamic Learning Style Prediction Method Based on a Pattern Recognition Technique,” *IEEE Transactions on Learning Technologies,* vol. 7, no. 2, pp. 165-77, 2014. |
| [2] | N. Othman dan M. H. Amiruddin, “Different perspectives of learning styles from VARK model,” *Procedia - Soc. Behav. Sci.,* vol. 7, no. 2, pp. 652-660, 2010. |
| [3] | W. A. Drago dan W. R. J, “VARK preferred learning styles and online education,” *Management Research News,* vol. 27, no. 7, pp. 1-13, 2004. |
| [4] | R. M. Felder dan L. K. Silverman, “Learning and Teaching Styles in Engineering Education,” *Engineering Education,* vol. 78, no. 7, pp. 674-681, 1988. |
| [5] | N. Fleming dan D. Baume, “Learning Styles Again: VARKing up the right tree!,” *Educational Developments, SEDA,* vol. 7, no. 4, pp. 4-7, 2006. |
| [6] | Bernard, Jason, T. W. Chang, E. Popescu dan S. Graf, “Learning Style Identifier: Improving the Precision of Learning Style Identification through Computational Intelligence Algorithms,” *Expert Systems with Applications,* vol. 75, pp. 94-108, 2017. |
| [7] | M. S. Hasibuan, L. E. Nugroho, P. I. Santosa dan S. S. Kusumawardani, “A Proposed Model for Detecting Learning Styles Based on Agent Learning,” *International Journal of Emerging Technologies in Learning,* vol. 11, no. 10, pp. 65-69, 2016. |
| [8] | Q. D. Pham dan A. M. Florea, “A Method for Detection of Learning Styles in Learning Management Systems,” *UPB Sci. Bull. Ser. C Electr. Eng.,* vol. 75, no. 4, pp. 3-12, 2013. |
| [9] | T. Hamtini dan H. Ateia, “A Proposed Dynamic Technique for Detecting Learning Style Using Literature Based Approach,” dalam *IEEE Jordan Conference on Applied Electrical Engineering and Computing Technologies (AEECT)*, 2015. |
| [10] | D. A. Kolb, Individual Learning Styles and The Learning Process, Sloan School of Management, Massachusetts Institue of Technology, 1971. |
| [11] | M. Radakovi, Balance between formal and informal learning - experience and challenges of civil servants training in Serbia, 2008. |
| [12] | Z. Pozgaj, “Informal Learning in Lifelong Education,” *International Journal of Emerging Technologies in Learning (iJET),* vol. 3, p. 46–50, 2008. |
| [13] | S. Graf, “Advanced Adaptivity in Learning Management Systems by Considering Learning Styles,” dalam *IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*, 2009. |
| [14] | J. D. Houwer, D. Barnes-Holmes dan A. Moors, “What is learning? On the nature and merits of a functional definition of learning,” *Psychonomic bulletin & review,* vol. 20, no. 4, pp. 631-642, 2013. |
| [15] | A. Barron, E. Hebets, T. Cleland, C. Fitzpatrick, M. Hauber dan J. Stevens, “Embracing multiple definitions of learning,” *Trends in neurosciences,* vol. 38, no. 7, pp. 405-407, 2015. |
| [16] | M. Domjan, The Essentials of Conditioning and Learning 5th Edition, Wadsworth/Thomson Learning, 2006. |
| [17] | G. Hall, “Learning, Psychology of,” dalam *Encyclopedia of Cognitive Science, L. Nadel (Ed.)*, Nature Publishing Group, 2003, pp. 837-845. |
| [18] | J. Gallacher, Second International Handbook of Lifelong Learning, 2013. |
| [19] | T. Ono, “Learning and Memory,” dalam *Encyclopedia Neuroscience*, 2009, p. 2129–2137. |
| [20] | N. Fleming, “VARK, a Guide to Learning Style,” VARK Learn Limited, 2018. [Online]. Available: http://vark-learn.com/. [Diakses 8 Maret 2018]. |
| [21] | Statista, Inc., “Number of smartphone users worldwide from 2014 to 2020 (in billions),” Statista, Juni 2016. [Online]. Available: https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/. [Diakses 5 Juni 2018]. |
| [22] | A. Fendelman, “How Are Cellphones Different From Smartphones?,” Lifewire, 17 Mei 2018. [Online]. Available: https://www.lifewire.com/cell-phones-vs-smartphones-577507. [Diakses 4 Juni 2018]. |
| [23] | Techopedia Inc., “Mobile Application (Mobile App),” Techopedia Inc., [Online]. Available: https://www.techopedia.com/definition/2953/mobile-application-mobile-app. [Diakses 4 Juni 2018]. |
| [24] | P. Viswanathan, “What Is a Mobile Application?,” Lifewire, 13 November 2017. [Online]. Available: https://www.lifewire.com/what-is-a-mobile-application-2373354. [Diakses 4 Juni 2018]. |
| [25] | Statista, Inc., “Number of mobile app downloads worldwide in 2017, 2018 and 2022 (in billions),” Mei 2018. [Online]. Available: https://www.statista.com/statistics/271644/worldwide-free-and-paid-mobile-app-store-downloads/. [Diakses 7 Juni 2018]. |
| [26] | M. Karch, “What Is Google Android?,” Lifewire, 2 Oktober 2017. [Online]. Available: https://www.lifewire.com/what-is-google-android-1616887. [Diakses 5 Juni 2018]. |
| [27] | R. Amadeo, “Google’s iron grip on Android: Controlling open source by any means necessary,” Arstechnica, 21 Oktober 2013. [Online]. Available: https://arstechnica.com/gadgets/2013/10/googles-iron-grip-on-android-controlling-open-source-by-any-means-necessary/. [Diakses 6 Juni 2018]. |
| [28] | S. Hill, “Android is losing its battle with fragmentation, and you’re paying the price,” Digital Trends, 2 November 2016. [Online]. Available: https://www.digitaltrends.com/mobile/what-is-android-fragmentation-and-can-google-ever-fix-it/. [Diakses 5 Juni 2018]. |
| [29] | IDC, “Smartphone OS,” IDC, Mei 2017. [Online]. Available: https://www.idc.com/promo/smartphone-market-share/os. [Diakses 6 Juni 2018]. |
| [30] | Google Developers, “UsageStatsManager,” Google Developers, [Online]. Available: https://developer.android.com/reference/android/app/usage/UsageStatsManager. [Diakses 8 Juni 2018]. |
| [31] | A. Fowler, NoSQL For Dummies, New Jersey: John Wiley & Sons, Inc, 2015. |
| [32] | JSON.org, “Introducing JSON,” JSON.org, [Online]. Available: http://json.org/. [Diakses 2 Juni 2018]. |
| [33] | Ecma International, The JSON Data Interchange Syntax, Ecma International, 2017. |
| [34] | K. Schwaber dan J. Sutherland, The Scrum Guide, Scrum Org., 2013. |
| [35] | P. Sahoo, Probability and Mathematical Statistics, Louisville, Kentucky: University of Louisville, 2013. |
| [36] | J. Cohen, Statistical Power Analysis for behavioral Science, 2nd ed., New Jersey: Routledge, 1998. |

LAMPIRAN

* *Test Case* Pengujian Skenario Aktivitas

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Skenario | Kasus Pengujian | Kebutuhan | No | Langkah |
| 1 | *Way to Point* | Autentikasi pengguna | Aplikasi telah dibuka | 1 | Memberikan izin akses kamera, lokasi, dan penyimpnanan |
|  | 2 | Memilih jenis akun EO atau *attendy* |
| Belum mempunyai akun | 3 | Masukkan data surel dan kata sandi |
|  | 4 | Menekan tombol *register* untuk mendaftarkan akun |
|  | 5 | Masukkan surel dan kata sandi untuk masuk ke aplikasi |
| Masukkan data peta baru, posisi acuan, data wifi | Aplikasi telah dibuka (EO) | 1 | Memilih tombol *Way to Point* |
| Belum ada peta yang dibuat | 2 | Memilih tombol tambah |
|  | 3 | Masukkan nama peta baru |
| Belum ada posisi acuan yang dibuat | 4 | Memilih tombol wifi *friendly* |
| Masuk ke halaman wifi *manager* | 5 | Memilih akses poin |
| Aktifkan wifi | 6 | Menekan tombol simpan |
| Aktifkan kamera | 7 | Memilih tombol *position* |
|  | 8 | Menggeser kamera ke kanan/kiri |
| *Progress bar* mencapai angka 100 | 9 | Memilih arah panah |
| Proses RSSI *fingerprinting* | 10 | Menekan tombol simpan |
| Menghapus daftar peta, posisi acuan, wifi | Aplikasi telah dibuka (EO) | 1 | Menggeser *item* yang ingin dihapus ke kanan atau kiri |
| 2 | *Way to Locate* | Mengkalkulasi posisi pengguna saat ini | Aplikasi telah dibuka | 1 | Memilih tombol *Way to Locate* |
| Aktifkan wifi & kamera | 2 | Menekan tombol *Locate Me* |
| Proses WKNN | 3 | Menunggu 10 detik |
| Proses SLAM | 4 | Menggeser kamera ke kanan/kiri |
| 3 | *Invitation* | Mengundang pengguna lain melalui surel | Aplikasi telah dibuka | 1 | Memilih tombol *Invitation* |

* *Test Case* Pengujian Skenario Aktivitas (Lanjutan)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Skenario | Kasus Pengujian | Kebutuhan | No | Langkah |
| 3 | *Invitation* | Mengundang pengguna lain melalui surel |  | 2 | Masukkan data nama acara, lokasi, tanggal, dan waktu |
|  | 3 | Memilih surel pengirim |
|  | 4 | Memilih surel penerima |
|  | 5 | Menekan tombol kirim |
| Mengundang pengguna lain melalui SMS | Aplikasi telah dibuka | 1 | Memilih tombol *Invitation* |
|  | 2 | Masukkan data nama acara, lokasi, tanggal, dan waktu |
| Firebase *invite* | 3 | Memilih surel pengirim |
|  | 4 | Memilih nomor kontak penerima |
|  | 5 | Menekan tombol kirim |

* Hasil Pengujian Akurasi Aplikasi terhadap Jumlah Akses Poin

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Percobaan ke - | Jumlah akses poin | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 1 | X | X | V | V | V | Posisi 2 | Posisi 2 | V | V | V |
| 2 | X | X | Posisi 2 | Posisi 2 | V | V | Posisi 2 | V | V | V |
| 3 | X | X | Posisi 2 | Posisi 2 | V | V | V | V | V | V |
| 4 | X | X | Posisi 2 | Posisi 2 | Posisi 2 | V | V | V | V | V |
| 5 | X | X | Posisi 2 | Posisi 2 | Posisi 2 | V | V | V | V | V |
| 6 | X | X | V | Posisi 2 | Posisi 2 | V | V | V | V | V |
| 7 | X | X | Posisi 2 | V | V | V | V | V | v | V |
| 8 | X | X | Posisi 2 | V | V | V | V | V | V | V |
| 9 | X | X | Posisi 4 | V | V | V | V | V | V | V |
| 10 | X | X | Posisi 4 | V | V | V | V | V | V | V |
| 11 | X | X | Posisi 4 | V | V | V | V | V | V | V |
| 12 | X | X | Posisi 2 | Posisi 2 | V | V | V | V | V | V |
| 13 | X | X | Posisi 2 | Posisi 2 | Posisi 2 | V | V | V | V | V |
| 14 | X | X | Posisi 4 | Posisi 2 | V | V | V | V | V | V |
| 15 | X | X | Posisi 2 | Posisi 2 | V | V | V | V | V | V |

* Hasil Pengujian Akurasi Aplikasi terhadap Jumlah Akses Poin (Lanjutan)

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Percobaan ke - | Jumlah akses poin | | | | | | | | | |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 16 | X | X | V | Posisi 2 | V | V | V | V | V | V |
| 17 | X | X | V | Posisi 2 | Posisi 2 | V | V | V | Posisi 4 | V |
| 18 | X | X | Posisi 2 | Posisi 4 | V | V | V | V | V | V |
| 19 | X | X | V | Posisi 4 | V | V | V | Posisi 4 | V | V |
| 20 | X | X | V | Posisi 2 | V | V | V | V | v | V |
| Akurasi (%) | 0 | 0 | 30 | 30 | 75 | 95 | 90 | 95 | 95 | 100 |
| Keterangan : X = Gagal, V = Berhasil (Posisi sesuai), Posisi x = Menunjukan posisi yang salah | | | | | | | | | | |