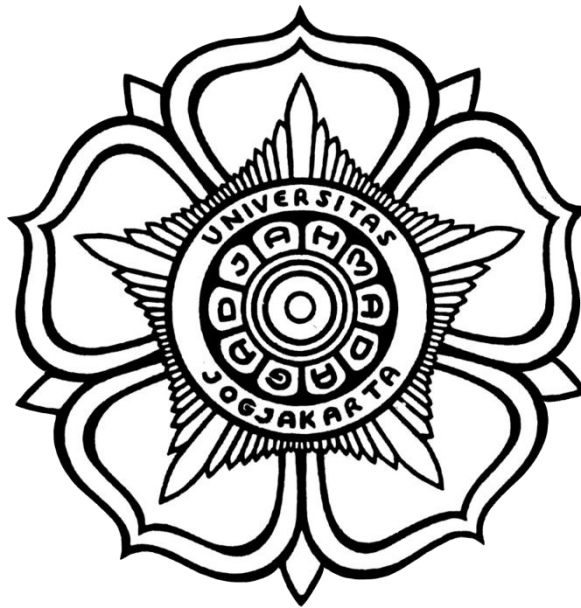


**ANALISIS UJI KORELASI ANTARA DATA PENGGUNAAN
APLIKASI SMARTPHONE TERHADAP MODEL GAYA BELAJAR
VARK (VISUAL, AURAL, READ/WRITE, KINESTHETIC)**

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
2018**

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 SHELDTY FERDINAND ERWIANDA

14/365238/TK/42057

Telah disetujui dan disahkan

pada tanggal <tanggal lulus ujian pendadaran>

Dosen Pembimbing I

Dosen Pembimbing II

P. Insap Santosa, Ir., M.Sc., Ph.D.
NIP. 196101081985031002

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 kepada Tuhan Yang Maha Esa, sumber dari segala ilmu dan pengetahuan, sebagai wujud rasa syukur telah diberikannya setetes ilmu dan pengetahuan kepadaku

KATA PENGANTAR

Puji syukur penulis haturkan ke hadirat Allah SWT Yang Maha Pengasih dan Maha Penyayang atas limpahan rahmat, karunia, hidayah, serta ridhoNya sehingga tugas akhir berupa penelitian dan penyusunan skripsi ini telah terselesaikan dengan baik.

Dalam penyusunan tugas akhir ini penulis telah banyak mendapatkan arahan, bantuan, serta dukungan dari berbagai pihak. Oleh karena itu penulis ingin 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. P. Insap Santosa, Ir., M.Sc., Ph.D. selaku dosen pembimbing pertama yang telah banyak memberikan ilmu, bimbingan, saran, dan pengertian kepada penulis.
3. Ibu Sri Suning Kusumawardani, Dr., S.T., M.T., selaku dosen pembimbing kedua yang juga telah banyak memberikan ilmu, bimbingan, saran, dan pengertian kepada penulis.
4. Bapak Sujoko Sumaryono, Ir., M.T. selaku dosen pembimbing akademik atas nasihat dan bimbingannya selama masa perkuliahan dari awal hingga akhir.
5. Bapak Warsun Najib, S.T., M.Sc. selaku dosen pembimbing kerja praktik atas nasihat dan bimbingannya selama masa kerja praktik.

6. Bapak Bimo Sunarfri Hantono, S.T., M.Eng., selaku koordinator tim penelitian bidang deteksi gaya belajar otomatis atas ide, saran, dan masukan yang diberikan.
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 Tamami dan Ibu Lailia Askurina yang telah mendidik dan membimbing penulis dengan baik serta atas doa restunya sehingga penulis dapat menggapai banyak hal selama perkuliahan ini
9. Adik penulis, Attalia Haura Sauzan yang selalu dapat memberikan keceriaan dan kebahagiaan kepada penulis.
10. Kepada dirinya yang tidak dapat disebutkan namanya. Terimakasih atas dukungan yang tiada henti diberikan kepada penulis serta tak pernah lupa mengingatkan penulis untuk selalu bahagia.
11. Kepada Ridho dan Niko, terimakasih atas pengalamannya selama kerja praktik di Project Child Indonesia.
12. Kepada rekan-rekan seperjuangan di Gadjah Mada Aerospace Team UGM, terima kasih atas kesempatan dan pengalaman luar biasa yang diberikan kepada penulis.
13. Kepada Tim PKM REN-Locator, Mas Sony, Royan, dan Fiha, terima kasih atas pengalaman luar biasa dalam berjuang di berbagai kompetisi hingga membuahkan hasil yang manis.

14. Kepada 21 manusia luar biasa yang senantiasa ada dikala suka maupun duka, terima kasih atas dukungannya selama masa kuliah ini.
15. Kepada kolega di Lab. Informatika serta laboran, terima kasih atas waktu dan fasilitas yang diberikan sehingga proses pembuatan tugas akhir ini berjalan dengan lancar.
16. Kepada teman-teman KKN-PPM Desa Jambu, Kabupaten Semarang, terima kasih atas pengalamannya dalam bekerja di dalam masyarakat. Terima kasih juga atas kesempatan luar biasa yang diberikan kepada penulis
17. Para ilmuwan dan peneliti di bidang deteksi gaya belajar, serta para responden yang telah bersedia meluangkan waktunya. Tanpa kalian maka tugas akhir ini tidak akan dapat terselesaikan.

Yogyakarta, 26 Juni 2018

Penulis

DAFTAR ISI

HALAMAN JUDUL.....	i
HALAMAN PENGESAHAN.....	ii
LEMBAR PERNYATAAN	iii
HALAMAN PERSEMBAHAN	v
KATA PENGANTAR	vi
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xii
DAFTAR GAMBAR	xv
DAFTAR SINGKATAN	xviii
INTISARI.....	xix
<i>ABSTRACT</i>	xx
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	4
1.3. Keaslian Penelitian.....	4
1.4. Batasan Masalah	7
1.5. Tujuan Penelitian	7
1.6. Manfaat Penelitian	8
1.7. Sistematika Penulisan	8
BAB II TINJAUAN PUSTAKA DAN DASAR TEORI.....	10
2.1. Tinjauan Pustaka	10
2.2. Dasar Teori.....	13
2.2.1. Belajar	13

2.2.2.	Model Gaya Belajar VARK	15
2.2.3.	<i>Smartphone</i>	16
2.2.4.	Aplikasi Peranti Bergerak	18
2.2.5.	Android	19
2.2.6.	App Usage Statistic API	21
2.2.7.	Firebase	22
2.2.8.	Evernote Android Job	24
2.2.9.	Metode Pengembangan Perangkat Lunak SCRUM.....	24
2.2.10.	Uji Korelasi Spearman	24
2.3.	<i>Pertanyaan Penelitian</i>	26
BAB III METODE PENELITIAN.....		26
3.1.	Alat dan Bahan.....	26
3.1.1.	Alat.....	26
3.1.2.	Bahan	28
3.2.	Alur Penelitian	29
3.2.1.	Identifikasi Masalah	30
3.2.2.	Pengembangan Aplikasi.....	31
3.2.3.	Pengumpulan Data Responden	32
3.2.4.	Uji Korelasi Data Penggunaan Aplikasi	33
3.2.5.	Penulisan Laporan.....	33
3.3.	Perancangan <i>Sprint</i>	33
3.4.	Prosedur Analisis Uji Korelasi.....	37
3.4.1.	Analisis Kebutuhan Data	38
3.4.2.	Pengambilan Data	38
3.4.3.	Pengolahan Data	39
3.4.4.	Analisis Uji Korelasi Spearman dan Penarikan Kesimpulan.....	40

BAB IV HASIL DAN PEMBAHASAN	42
4.1. Pengembangan Aplikasi <i>Usage Statistic Recorder</i>	42
4.1.1. <i>Sprint 1</i> : Layanan Pencatat Data Penggunaan Aplikasi	43
4.1.2. <i>Sprint 2</i> : Fitur Pengisian Kuesioner VARK	48
4.1.3. <i>Sprint 3</i> : Persetujuan Pengguna, Petunjuk Penggunaan Aplikasi, dan Pengisian Identitas Responden.....	55
4.1.4. <i>Sprint 4</i> : Sinkronisasi Firebase, Autentikasi, dan Penyelesaian Akhir	59
4.2. Analisis Uji Korelasi	69
4.2.1. Pengolahan Awal Data	71
4.2.2. Analisis Uji Korelasi Skenario 1	73
4.2.3. Analisis Uji Korelasi Skenario 2	88
4.2.4. Analisis Uji Korelasi Skenario 3	95
BAB V KESIMPULAN DAN SARAN.....	107
51.1. Kesimpulan	107
51.1. Saran	108
DAFTAR PUSTAKA	109
LAMPIRAN	134

DAFTAR TABEL

Tabel 1.1. Perbandingan aspek-aspek penelitian.....	6
Tabel 2.1. Perbandingan kategori proses pembelajaran	14
Tabel 3.1. Spesifikasi laptop 1 yang digunakan dalam penelitian	26
Tabel 3.2. Spesifikasi laptop 2 yang digunakan dalam penelitian	26
Tabel 3.3. Spesifikasi <i>smartphone</i> 1 yang digunakan dalam penelitian	27
Tabel 3.4. Spesifikasi <i>smartphone</i> 2 yang digunakan dalam penelitian	27
Tabel 3.5. Fitur dan kebutuhan aplikasi	33
Tabel 3.6. <i>Product Backlog</i> aplikasi	35
Tabel 3.7. <i>Sprint</i> 1: layanan penangkap data penggunaan aplikasi.....	35
Tabel 3.8. <i>Sprint</i> 2 : kuesioner VARK	36
Tabel 3.9. <i>Sprint</i> 3 : persetujuan pengguna, petunjuk penggunaan aplikasi dan pengisian identitas responden	36
Tabel 3.10. <i>Sprint</i> 4 : sinkronisasi Firebase, autentikasi, dan penyelesaian akhir .	37
Tabel 3.11. Tabel Perbandingan Penelitian-Penelitian Sebelumnya [7]	39
Tabel 4.1. <i>Sprint</i> 1 Overview	48
Tabel 4.2. <i>Sprint</i> 2 Overview	54
Tabel 4.3. Daftar <i>petunjuk</i> dan ketentuan survey	56
Tabel 4.4. <i>Sprint</i> 3 Overview	59
Tabel 4.5. <i>Sprint</i> 4 Overview	69
Tabel 4.6. Tabel Nilai VARK Aktual.....	70

Tabel 4.7. Tabel Data Penggunaan Aplikasi	71
Tabel 4.8. Hasil Pengolahan Awal terhadap Data Penggunaan Aplikasi.....	73
Tabel 4.9. Tabel <i>Dataset</i> Skenario 1	75
Tabel 4.10. Hasil Pengujian Durasi Penggunaan “com.adobe.lrmobile”	75
Tabel 4.11. Hasil Pengujian Durasi Penggunaan “com.android.calculator2”	76
Tabel 4.12. Hasil Pengujian Durasi Penggunaan “com.android.mms”	77
Tabel 4.13. Hasil Pengujian Durasi Penggunaan “com.duapps.recorder”	78
Tabel 4.14. Hasil Pengujian Durasi Penggunaan “com.microsoft.office.word”	79
Tabel 4.15. Hasil Pengujian Durasi Penggunaan “com.mobile.legends”	80
Tabel 4.16. Hasil Pengujian Durasi Penggunaan “com.pinterest”	81
Tabel 4.17. Hasil Pengujian Durasi Penggunaan “com.samsung.android.video” ..	82
Tabel 4.18. Hasil Pengujian Durasi Penggunaan “com.supercell.hayday”	83
Tabel 4.19. Hasil Pengujian Frekuensi Penggunaan “cn.wps.moffice_eng”	84
Tabel 4.20. Hasil Pengujian Frekuensi Penggunaan “com.android.camera”	85
Tabel 4.21. Hasil Pengujian Frekuensi Penggunaan “com.facebook.katana”	86
Tabel 4.22. Hasil Uji Frekuensi Penggunaan “com.android.quicksearchbox”	87
Tabel 4.23. Tabel Nilai Dominan VARK.....	89
Tabel 4.24. <i>Dataset</i> Pengujian Skenario 2	90
Tabel 4.25. Uji Durasi “com.google.android.youtube” terhadap Kelas A	91
Tabel 4.26. Uji Durasi “com.microsoft.office.powerpoint” terhadap Kelas K	92
Tabel 4.27. Uji Frekuensi “com.android.phone” terhadap Kelas A	92

Tabel 4.28. Uji Frekuensi “com.google.android.apps.maps” terhadap Kelas K....	94
Tabel 4.29. Tabel Hasil Pengolahan Data Tahap Kedua.....	96
Tabel 4.30. Tabel Rincian Kategori Aplikasi.....	97
Tabel 4.31. Pengujian Durasi “AUDIO_PLAYER”	99
Tabel 4.32. Hasil Uji Korelasi pada Durasi “CALCULATOR”	100
Tabel 4.33. Tabel Hasil Uji Korelasi pada Durasi “DICTIONARY”	101
Tabel 4.34. Tabel Hasil Uji Korelasi pada Durasi “EDUCATION”	102
Tabel 4.35. Tabel Hasil Uji Korelasi pada Durasi “GAME_SIMULATION”	103
Tabel 4.36. Tabel Hasil Uji Korelasi pada Durasi “VISUAL_EDITOR”	104
Tabel 4.37. Tabel Hasil Uji Korelasi pada Frekuensi “COMMUNICATION” ...	105
Tabel 4.38. Hasil Uji Korelasi pada Frekuensi “WEATHER”	106

DAFTAR GAMBAR

Gambar 2.1.	Relasi antara komponen-komponen penelitian	12
Gambar 2.2.	Grafik jumlah pengguna <i>smartphone</i> [21]	17
Gambar 2.3.	Grafik jumlah pengunduhan aplikasi peranti bergerak [25].....	19
Gambar 2.4.	Grafik pasar <i>smartphone</i> berdasarkan sistem operasi [27].....	20
Gambar 2.5.	Grafik pasar sistem operasi <i>smartphone</i> tahun 2014-2017 [29].....	21
Gambar 2.6.	Contoh hasil masukan dan keluaran <i>App Usage Statistic</i> API.....	22
Gambar 2.7.	Contoh basis data tidak terstruktur	23
Gambar 3.1.	Diagram alir penelitian	30
Gambar 3.2.	Metode pengembangan aplikasi SCRUM	31
Gambar 3.3.	Diagram alir prosedur analisis uji korelasi.....	37
Gambar 4.1.	Urutan kerja aplikasi.....	42
Gambar 4.2.	Contoh kode untuk memberikan izin akses API	43
Gambar 4.3.	Menu akses pintas untuk membuka pengaturan perangkat : a) Tampilan antarmuka pada aplikasi; b) Tampilan menu pengaturan izin penggunaan API.....	44
Gambar 4.4.	Contoh kode program untuk melakukan <i>query</i> UsageStats	45
Gambar 4.5.	Contoh kode program untuk mengatur beginTime dan endTime ..	45
Gambar 4.6.	Contoh kode program untuk melakukan <i>query</i> UsageEvents	46
Gambar 4.7.	Kelas MyJob berjenis Android Job	47
Gambar 4.8.	Contoh <i>log</i> data durasi dan frekuensi penggunaan aplikasi	47

Gambar 4.9. Tampilan antarmuka pengisian kuesioner VARK.....	49
Gambar 4.10. Sumber data pertanyaan dan jawaban survey VARK yang tersimpan di Firebase	50
Gambar 4.11. POJO kelas QuestionAndAnswers	50
Gambar 4.12. Contoh kode program untuk mengacak urutan pilihan jawaban	51
Gambar 4.13. Contoh kode program untuk mengambil data pertanyaan dan jawaban dari Firebase.....	51
Gambar 4.14. Hasil pengacakan urutan pertanyaan dan jawaban survey: a) Aplikasi dijalankan pertama kali; b) Aplikasi dijalankan kedua kali	52
Gambar 4.15. Contoh kode program untuk menghitung hasil survey.....	52
Gambar 4.16. Contoh <i>log</i> data hasil survey VARK	53
Gambar 4.17. Tampilan halaman akhir	53
Gambar 4.18. Contoh kode program untuk keluar dari aplikasi	54
Gambar 4.19. Tampilan persetujuan pengguna dan petunjuk survey	55
Gambar 4.20. Contoh kode program untuk keluar dari aplikasi	56
Gambar 4.21. Halaman pengisian identitas responden	57
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	58
Gambar 4.23. Pengaturan autentikasi pengguna pada <i>dashboard</i> Firebase	60
Gambar 4.24. Contoh kode program proses autentikasi pengguna.....	61
Gambar 4.25. POJO kelas User.....	62
Gambar 4.26. Contoh kode program untuk menyimpan data pengguna.....	62

Gambar 4.27. Contoh hasil penyimpanan data pengguna	63
Gambar 4.28. Kelas POJO Result	64
Gambar 4.29. Contoh kode program untuk menyimpan data hasil survey	64
Gambar 4.30. Contoh hasil penyimpanan data survey ke basis data Firebase	64
Gambar 4.31. POJO kelas MyUsageStats	65
Gambar 4.32. Contoh kode program untuk menyimpan data ke Firebase	66
Gambar 4.33. Contoh hasil penyimpanan data ke Firebase	66
Gambar 4.34. Diagram alur kerja aplikasi	67
Gambar 4.35. Contoh kode program untuk memeriksa apakah pengguna sudah terautentikasi atau belum.....	68
Gambar 4.36. Contoh kode program untuk memeriksa apakah pengguna sudah pernah mengisi survey atau belum.....	68
Gambar 4.37. Ilustrsi Perubahan Struktur Data	70
Gambar 4.38. Kode pemrograman untuk pengolahan data tahap pertama	72
Gambar 4.39. Kode Pemrograman untuk Pengolahan Data pada Skenario 1	74
Gambar 4.40. Kode Pemrograman untuk Menentukan Kelas Paling Dominan....	88
Gambar 4.41. Pemrograman Pembentukan <i>Dataset</i> Skenario 2	90
Gambar 4.42. Kode Pemrograman untuk Pengelompokan Aplikasi.....	95
Gambar 4.43. Kode Pemrograman untuk Pembentukan Data Pengujian.....	98

DAFTAR SINGKATAN

A

AI	<i>Artificial Intelligence</i>
API	<i>Application Programming Interface</i>
AUS	<i>App Usage Statistics</i>

F

FSLSM	<i>Felder-Silverman Learning Style Model</i>
-------	--

I

IDE	<i>Integrated Development Environment</i>
-----	---

J

JSON	<i>JavaScript Object Notation</i>
------	-----------------------------------

N

NoSQL	<i>Not only Structured Query Language</i>
-------	---

S

SDK	<i>Standard Development Kit</i>
SQL	<i>Structured Query Language</i>

U

UI	<i>User Interface</i>
UID	<i>Unique Identifier</i>

V

VARCK	<i>Visual, Aural, Read/Write, Kinesthetic</i>
-------	---

X

XML	<i>Extensible Markup Language</i>
-----	-----------------------------------

INTISARI

Proses pembelajaran manusia berlangsung sepanjang hidup untuk mengembangkan atau mendapatkan keterampilan, baik itu secara formal, non-formal, maupun informal. Deteksi gaya belajar adalah hal yang dilakukan untuk mengetahui kecenderungan perilaku seseorang selama proses pembelajaran tersebut. Ada beberapa model gaya belajar yang dikembangkan. Salah satu model gaya belajar yang dapat digunakan adalah model VARK.

Smartphone menjadi salah satu media yang berpotensi untuk melakukan deteksi gaya belajar secara informal karena terdapat data penggunaan berbagai macam aplikasi yang terpasang di dalamnya. Namun, belum ada penelitian yang mencari tahu apakah data penggunaan aplikasi *smartphone* memiliki hubungan terhadap preferensi gaya belajar seseorang. Agar dapat melakukan pengujian korelasi tersebut, diperlukan kedua buah data. Untuk memudahkan hal tersebut, diperlukan sebuah aplikasi bantu yang berfungsi mengumpulkan data survey kuesioner dan data penggunaan aplikasi *smartphone*.

Hasil dari aplikasi yang telah dikembangkan, aplikasi tersebut mampu mengumpulkan kedua data tersebut serta menyimpannya ke basis data Firebase sehingga proses pengujian korelasi dapat berlangsung dengan mudah. Hasil dari pengujian korelasi, diperoleh bahwa durasi dan frekuensi penggunaan beberapa aplikasi memiliki korelasi yang signifikan terhadap model gaya belajar VARK. Selain itu, hasil pengkategorian aplikasi juga memiliki korelasi yang signifikan terhadap model gaya belajar VARK.

Kata kunci : gaya belajar VARK, uji korelasi, *app usage statistics*, pembelajaran informal

ABSTRACT

Learning is a long life process to improve or learn a new skill, whether it is formally, non-formally, or informally. Learning style detection is a process to understand how humans learn. There are some developed learning style model. One of them is VARK learning style model.

Smartphone has some valuable data about its application usage, making it becomes one of the potential medias to detect informal learning style preference. . Unfortunately, no research has been done to find out the correlation between learning style preference and smartphone's application usage. To facilitate this, it is necessary to create an application to obtain both data.

As the results, this applicaion is able to collect the results of learning style questionnaire and the appplication usage data as well as savings them in Firebase, so that the correlation test can be performed easily. From the correlation test, some applications' duration and frequency have significant correlation result with VARK learning style model. In addition, the categorization results of the application also has a significant correlation with VARK learning style model.

Keywords : *VARK learning style model, correlation test, app usage statistics, informal learning*

BAB I

PENDAHULUAN

1.1. 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.

1.2. 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 *smartphone* 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.

1.3. 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. Perbandingan aspek-aspek penelitian

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

1.4. 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*.

1.5. 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.

1.6. 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.

1.7. 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.

2. 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

3. 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.

4. 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.

5. 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

2.1. 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 maksud 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.

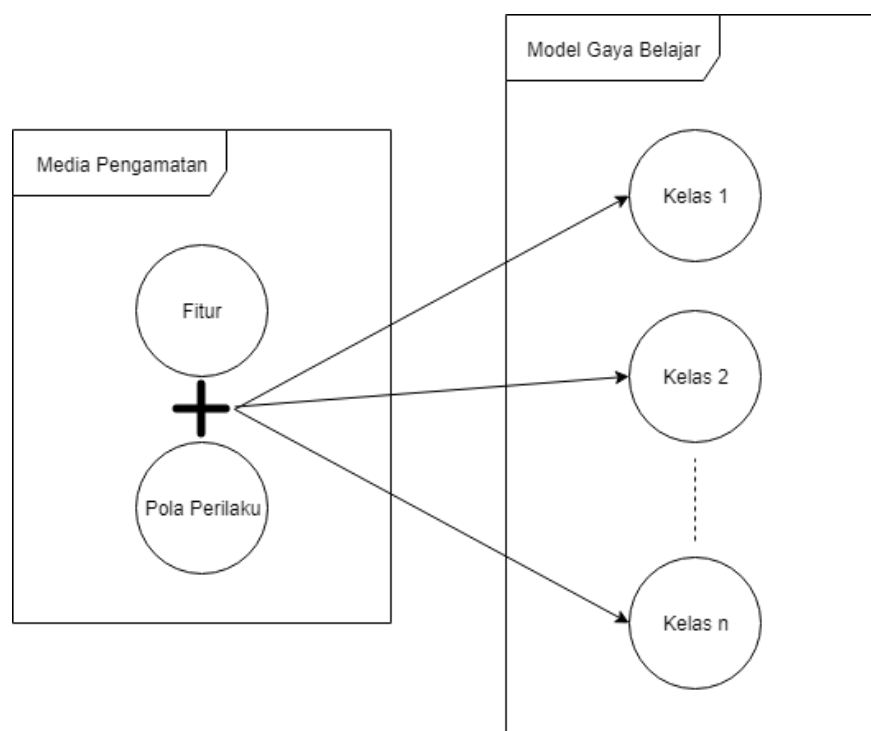
Pham *et.al* memaparkan metodologi untuk melakukan pengidentifikasian gaya belajar dengan pendekatan otomatis [8]. Terdapat 4 aspek utama yang perlu ditentukan dalam mengembangkan sistem pengidentifikasi gaya belajar otomatis, yaitu model gaya belajar acuan, media pengamatan, fitur, dan pola perilaku. 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 mengategorikan 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.

2.2. Dasar Teori

2.2.1. 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 penambahan 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. Kategori ketiga yaitu pembelajaran informal, merupakan pembelajaran 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. 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]:

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

2.2.2. 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]:

- a. *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.
- b. *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.

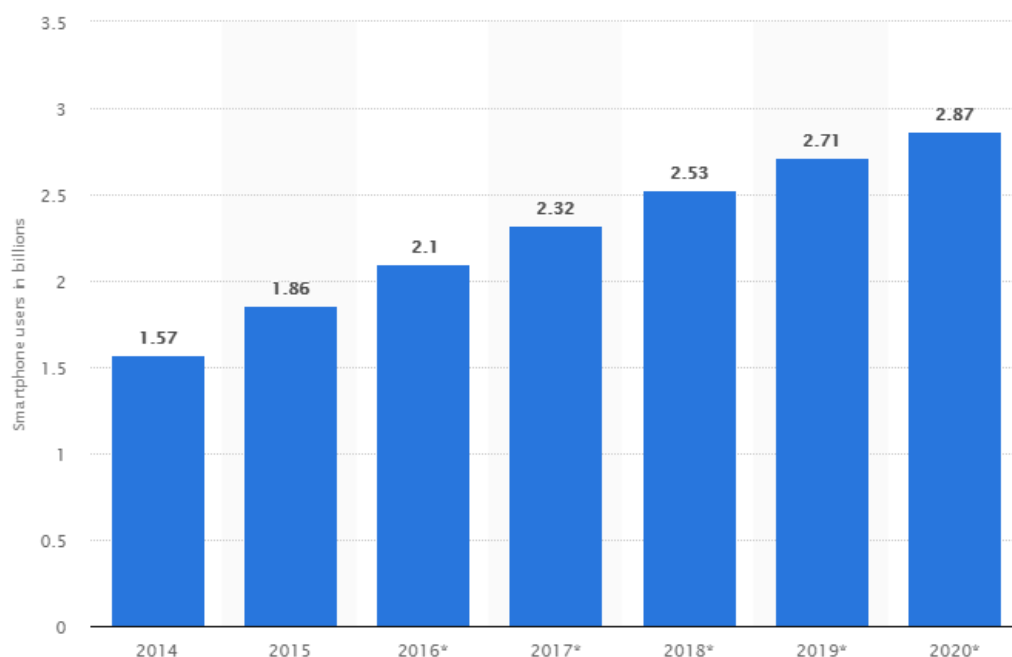
- c. *Read/Write*: preferensi ini mencakupi segala informasi yang disajikan secara tekstual.
- d. *Kinesthetic*: individu dengan preferensi ini cenderung memahami informasi melalui pengalaman, contoh nyata, melakukan praktek, demonstrasi, dan simulasi. Individu dengan preferensi ini cenderung banyak 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*.

2.2.3. *Smartphone*

Tidak terdapat definisi pasti dari *smartphone*, namun *smartphone* sering dianggap sebagai komputer berukuran kecil. Anggapan tersebut muncul karena *smartphone* memiliki 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]

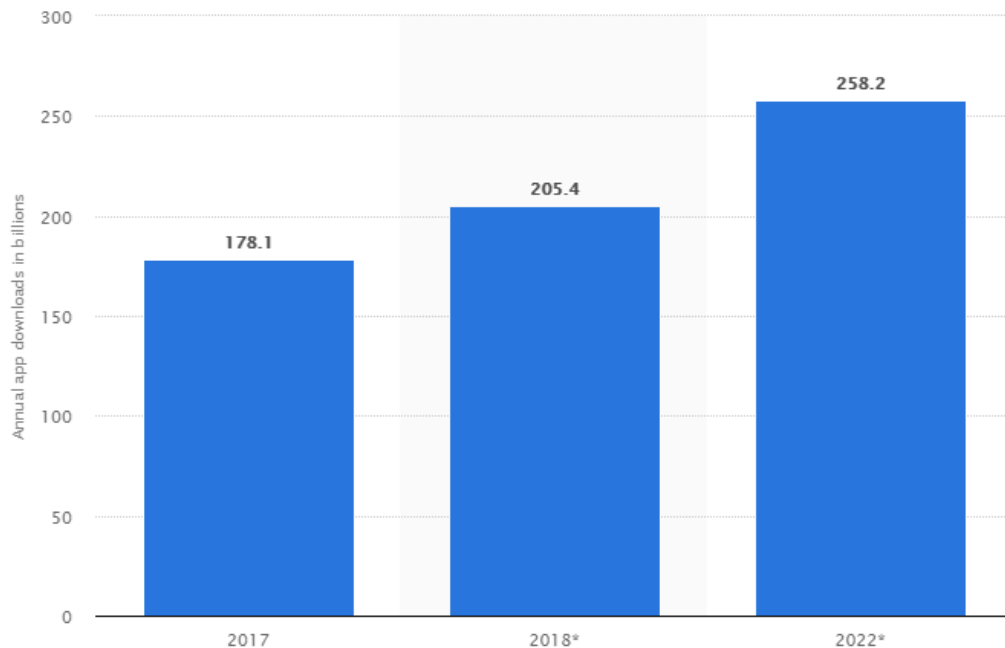
Garis batas yang tidak jelas membuat orang terkadang sulit membedakan *smartphone* dari telepon genggam konvensional, ataupun peranti komputer bergerak lain seperti tablet. Berdasarkan segi ukuran, *smartphone* umumnya seukuran dengan telepon genggam konvensional dan relatif jauh lebih kecil dari komputer 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].

2.2.4. 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 berkembang 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.



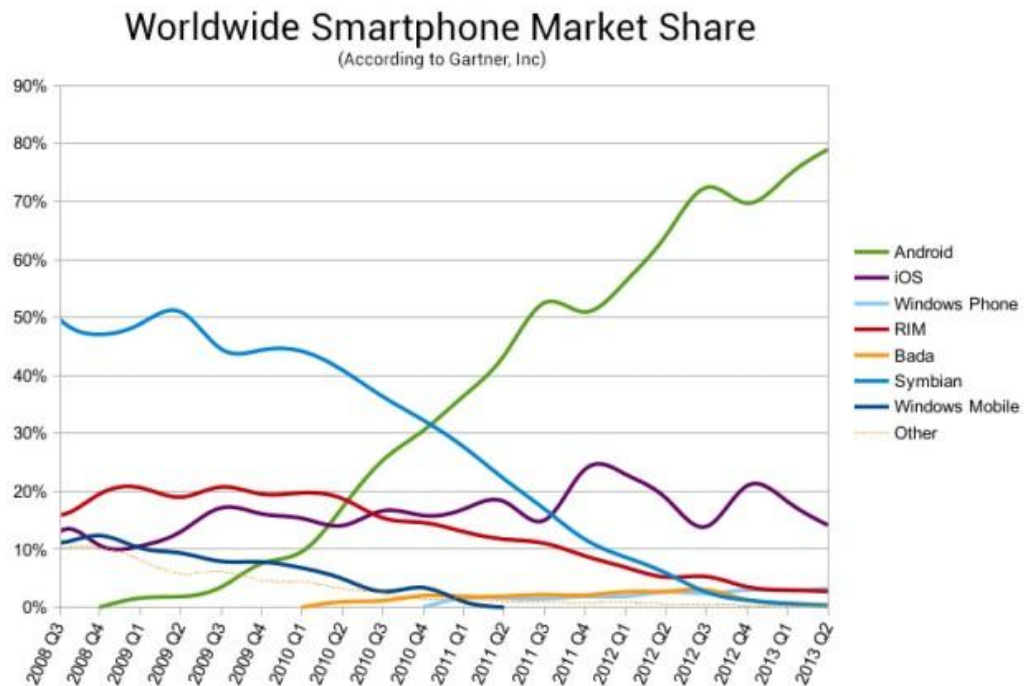
Gambar 2.3. Grafik jumlah pengunduhan aplikasi peranti bergerak [25]

2.2.5. 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.

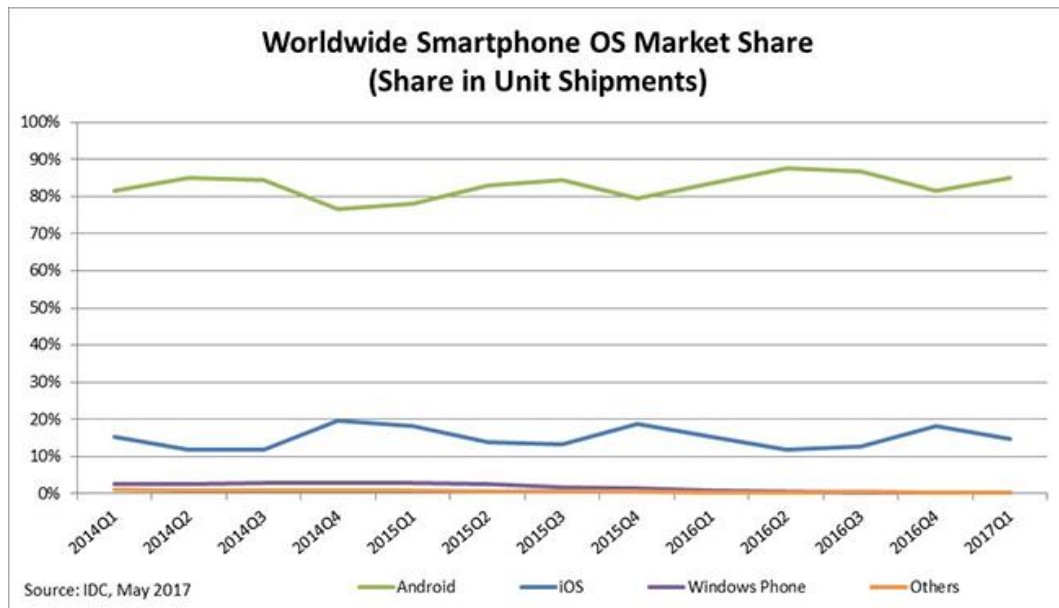
Android bersifat *open source*, sehingga banyak pihak ketiga yang dapat menggunakan *platform* Android secara gratis untuk dikembangkan sesuai kepentingan masing-masing [26]. Google juga membentuk suatu kelompok bernama Open Handset Alliance yang terdiri dari perusahaan-perusahaan perangkat keras, perangkat 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]

2.2.6. App Usage Statistic API

Mulai versi Lollipop (Android 5.0), Android menyediakan sebuah API (*Application 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:

- Android SDK 27
- Android Build Tools v27.0.2
- 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
```

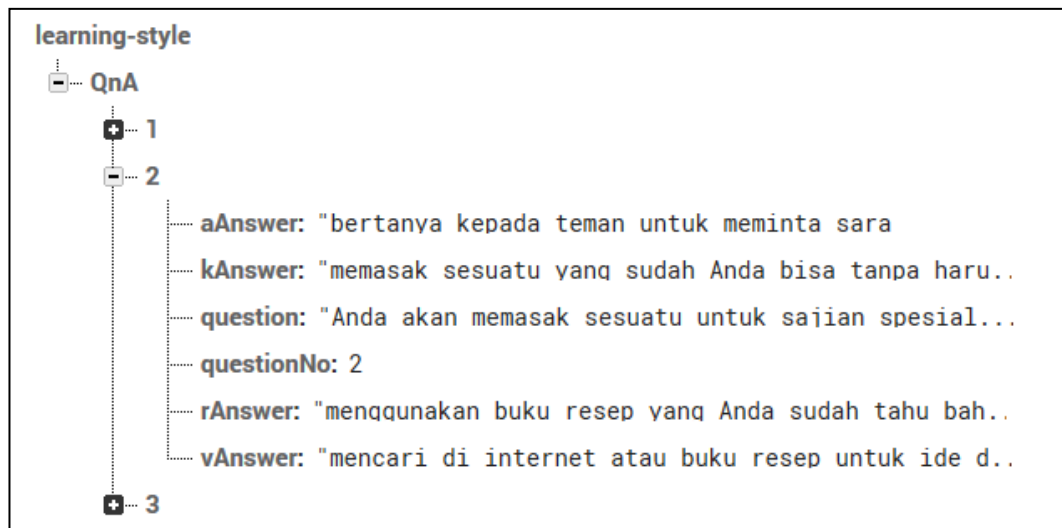
Gambar 2.6. Contoh hasil masukan dan keluaran *App Usage Statistic* API

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*.

2.2.7. 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 2.7. Contoh basis data tidak terstruktur

Firestore 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 2.7. Maka format basis data firestore 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, Firestore memiliki kelebihan dibandingkan basis data tidak terstruktur lainnya, yaitu Firestore mendukung layanan basis data secara *real time*.

2.2.8. 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.

2.2.9. 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].

2.2.10. Uji Korelasi Spearman

Uji korelasi spearman merupakan suatu metode untuk mencari tingkat keterkaitan di antara dua buah variabel dengan rumus berikut [35]

Tingkat ketergantungan/dependensi dari 2 peubah x dan y dapat dihitung untuk menghasilkan suatu nilai ketergantungan. Nilai ketergantungan tersebut disebut sebagai koefisien korelasi. Koefisien korelasi antara dua peubah x dan y dapat dihitung dengan rumus [36]:

$$\rho = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2 \sum_i (y_i - \bar{y})^2}} \quad (1)$$

dengan:

p : koefisien korelasi

x : nilai peubah pertama

y : nilai peubah kedua

n : banyaknya sampel

Koefisien korelasi menngambarkan seberapa kuat keterkaitan di antara 2 peubah yang diamati. Koefisien korelasi selalu terletak pada rentang [-1,1]. Jika koefisien korelasi bernilai 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 koefisien korelasi bernilai 1, maka kedua peubah x dan y memiliki korelasi sempurna. Nilai positif pada koefisien korelasi berarti semakin tinggi nilai x, maka semakin tinggi pula nilai y, dan sebaliknya. Jika koefisien korelasi bernilai -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 lima kelompok, yaitu [37]:

- a. 0-0,19 = sangat lemah
- b. 0,2-0,39 = lemah
- c. 0,4-0,59 = menengah
- d. 0,6-0,79 = kuat
- e. 0,8-1 = sangat kuat

2.3. Pertanyaan Penelitian

1. Apakah aplikasi yang dikembangkan dapat menangkap data durasi dan frekuensi penggunaan aplikasi serta data hasil pengisian kuesioner VARK?
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

3.1. Alat dan Bahan

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

3.1.1. Alat

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

a. 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
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

b. Perangkat Lunak

Beberapa perangkat lunak yang digunakan untuk mengembangkan aplikasi adalah:

1. Android Studio

Android Studio adalah sebuah *Integrated Development Environment* (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.

2. 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*.

3. 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.

4. 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 visualisasi data dalam bentuk tabel.

5. 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.

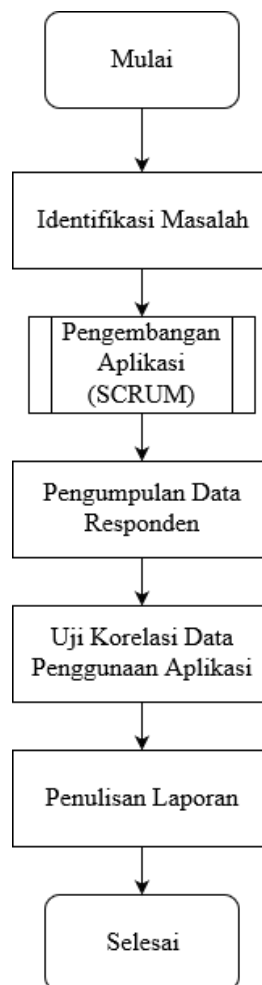
3.1.2. Bahan

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.

3.2. 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.



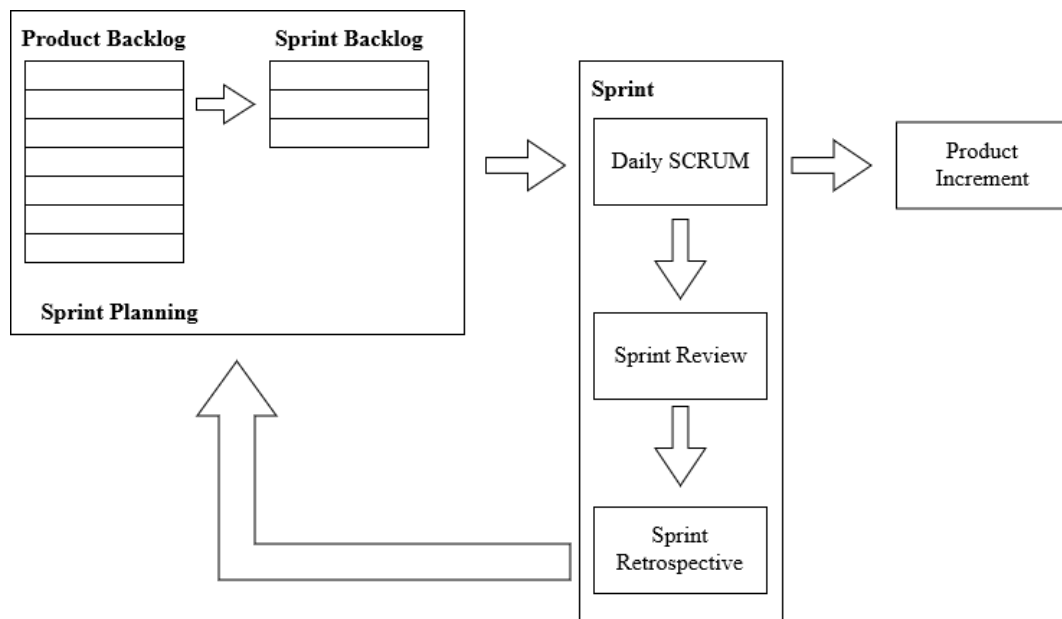
Gambar 3.1. Diagram alir penelitian

3.2.1. 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 memperkuat permasalahan dan mendasari penelitian ini dilakukan. Referensi yang digunakan bersumber dari buku, jurnal, artikel, dan situs-situs internet.

3.2.2. 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

a. 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.

b. 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.

c. 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.

d. Sprint Review

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

e. 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.

f. 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*.

3.2.3. 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.

3.2.4. 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.

3.2.5. Penulisan Laporan

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

3.3. 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	Aplikasi mampu menampilkan 16 pertanyaan dari kuesioner VARK beserta 4 pilihan jawaban

Fitur	Detail	Kebutuhan
	VARK melalui aplikasi	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 <i>smartphone</i> responden	Layanan dapat menangkap data historis durasi penggunaan aplikasi
		Layanan dapat menangkap data historis frekuensi penggunaan aplikasi
		Layanan harus dapat berjalan di balik layar (<i>background</i>)
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 setuju 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	Fitur-fitur yang telah	Pembuatan alur kerja aplikasi

Fitur	Detail	Kebutuhan
akhir aplikasi	dibuat digabungkan menjadi satu aplikasi utuh dan disusun berdasarkan 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 <i>App Usage Access</i>
	Mengembangkan layanan penangkap data durasi penggunaan aplikasi
	Mengembangkan layanan penangkap data frekuensi penggunaan aplikasi
	Membuat layanan berjalan pada <i>background</i>
	<i>Sprint overview</i>

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
	<i>Sprint overview</i>

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

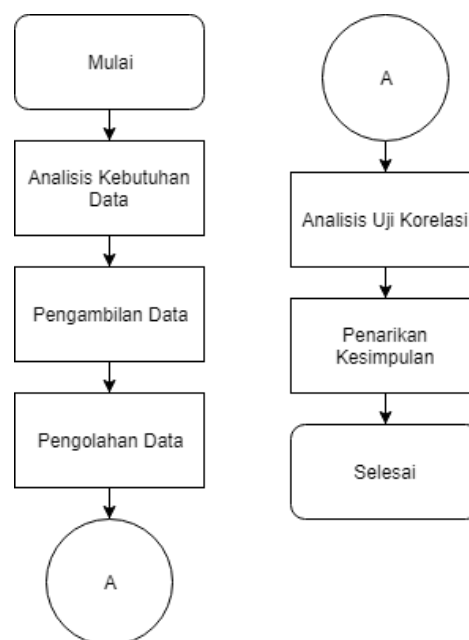
Prioritas	Product Backlog
3	Persetujuan Pengguna dan Petunjuk Penggunaan Aplikasi
	Penyusunan persetujuan pengguna dan petunjuk penggunaan aplikasi
	Mengembangkan tampilan persetujuan pengguna, petunjuk dan ketentuan survey
	Mengembangkan metode untuk mengecek apakah pengguna bersedia berpartisipasi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

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
	<i>Sprint overview</i>

3.4. 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

3.4.1. 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.

3.4.2. 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 VARK yang telah tersedia pada aplikasi perangkat lunak yang dikembangkan. Sedangkan pengumpulan data penggunaan aplikasi *smartphone* responden dilakukan menggunakan aplikasi piranti bergerak berbasis Android.

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

	E Ozpolat et al [3]	Garcia et al [4]	Ahmad et al [12]	PQ Dung et al [13]	Graf [2]
<i>Year</i>	2009	2007	2013	2012	2007
<i>Sample</i>	25 students	27 students	20 students	44 students	127 students
<i>Approach</i>	Data Driven	Data Driven	Literature Based	Literature Based	Literature Based
<i>Method</i>	NB Tree Classification	Bayesian	Behaviors Pattern	Behaviors	Behaviors Pattern
<i>Target Study</i>	Engineering Education	AI	Interactive Multimedia	All	Object Oriented Modelling
<i>Assessment Method</i>	Learning Model: ILS	BN Model: ILS	Behaviour: ILS	Learning Object: ILS	
<i>Precision</i>	67.7%		75%-83%		
<i>Dimension</i>					
Ac/Ref	70%	77%	75%	72.73%	79.33%
Sen/Int	73.33%	63%		70.15%	
Glo/Sec	73.33%	58%		79.54%	
Vis/Ver	53%	-		65.91%	

3.4.3. Pengolahan Data

Pengolahan data akan dilakukan dengan bahasa pemrograman Python. Secara garis besar, pengolahan data meliputi 3 tahapan berikut: pembersihan data, normalisasi data, dan pengelompokan data. Namun, penelitian ini akan menerapkan 3 skenario dalam analisis uji korelasi, sehingga juga terdapat 3 skenario pengolahan data yang berbeda. Titik perbedaan pengolahan data untuk tiap skenario terletak pada tahap pengelompokan data. Tahap pembersihan data merupakan tahap pertama pada semua skenario yang diterapkan. Tidak semua aplikasi bersifat relevan bagi penelitian ini. Terdapat sejumlah aplikasi yang tidak berkaitan dengan kelas-kelas VARK, seperti anti virus, alat konfigurasi sistem *smartphone*, alat ekstraksi berkas, dll. Selain itu, terdapat sejumlah baris data yang memiliki nilai 0 pada kedua kolom pengukuran yang akan diuji, yaitu kolom durasi dan frekuensi. Baris data dengan sifat seperti itu tidak dapat digunakan dalam analisis. Maka, perlu dilakukan pembersihan data dari aplikasi-aplikasi yang tidak relevan tersebut. Analisis uji korelasi juga dipengaruhi oleh jumlah sampel, maka dilakukan pembersihan pula bagi aplikasi-aplikasi dengan sampel yang terlalu sedikit. Batas sampel yang ditetapkan pada penelitian ini adalah 5 sampel.

Tahap kedua yang juga diterapkan pada semua skenario pengujian adalah normalisasi data. Normalisasi akan dilakukan terhadap data-data pada kedua kolom pola pengukuran. Normalisasi perlu dilakukan karena tiap responden

memiliki rentang nilai pola pengukuran yang berbeda-beda tergantung sifat penggunaannya terhadap *smartphone*. Normalisasi akan mengatasi masalah perbedaan rentang dengan menerapkan suatu fungsi untuk menyetarakan rentang nilai pengukuran ke dalam $[0...1]$. Adapun fungsi normalisasi yang diterapkan adalah sebagai berikut:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (2)$$

dengan:

x = nilai asli

x' = nilai ternormalisasi

Tahap terakhir dan merupakan letak titik perbedaan dari tiap skenario adalah pengelompokan data. Pengelompokan akan dilakukan terhadap aplikasi-aplikasi yang terekam. Skenario pertama tidak menerapkan tahap ini. Skenario 2 mengelompokkan aplikasi berdasarkan nilai VARK dominan dari penggunaannya. Sedangkan skenario 3 mengelompokkan aplikasi berdasarkan fungsinya. Pengelompokan pada skenario 3 dilakukan melalui 2 tahap. Tahap pertama merupakan tahap pengelompokan otomatis. Proses ini dilakukan dengan menggali data kategori aplikasi dari situs Google Play Store dan membubuhkannya bagi tiap aplikasi yang bersangkutan. Tidak semua aplikasi tersedia pada Google Play Store, maka diperlukan tahap kedua. Tahap kedua merupakan pengelompokan aplikasi secara manual berdasarkan fungsinya, sehingga semua aplikasi yang relevan memiliki kategori. Setelah melalui tahap akhir ini, barulah data siap digunakan dalam analisis uji korelasi.

3.4.4. Analisis Uji Korelasi Spearman dan Penarikan Kesimpulan

Analisis uji korelasi dilakukan dengan dukungan perangkat lunak SPSS. Analisis uji korelasi melibatkan 2 peubah untuk diamati. Analisis uji korelasi akan dilakukan dalam 3 skenario yang berbeda, sehingga memiliki 3 pasang peubah yang berbeda pula. Skenario pertama akan melakukan uji korelasi antara durasi dan frekuensi penggunaan dari tiap aplikasi terhadap nilai-nilai dari

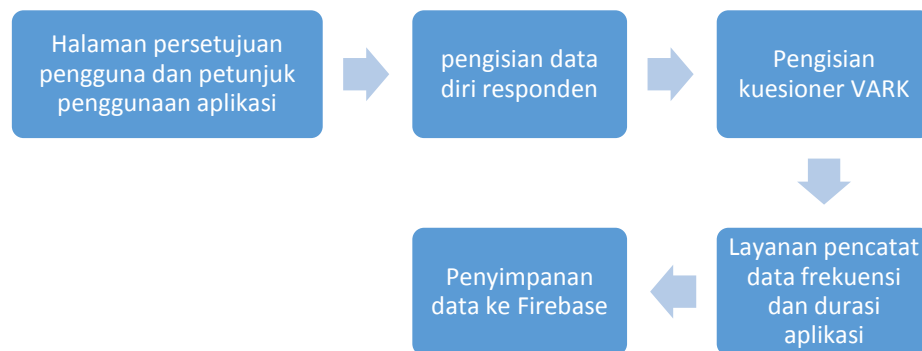
semua kelas VARK. Skenario kedua akan melakukan uji korelasi antara durasi dan frekuensi penggunaan dari tiap kelompok aplikasi yang dikategorikan berdasarkan nilai VARK dominan dari penggunanya, terhadap nilai VARK aktual pada kelas yang sama. Skenario ketiga akan melakukan uji korelasi antara durasi dan frekuensi penggunaan dari tiap kelompok aplikasi yang dikategorikan berdasarkan fungsinya, terhadap nilai-nilai dari semua kelas VARK.

BAB IV

HASIL DAN PEMBAHASAN

4.1. 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.

4.1.1. *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.

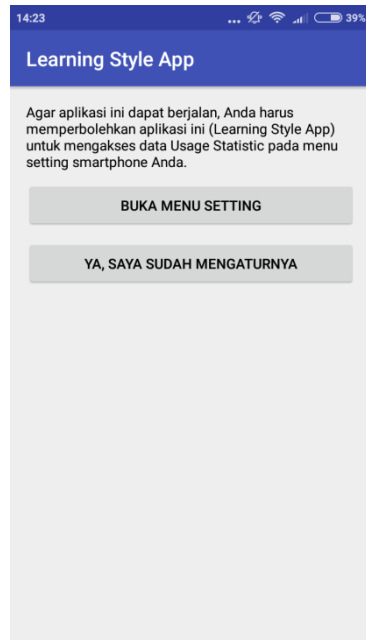
Langkah pertama yang dilakukan adalah memperbolehkan akses aplikasi terhadap API tersebut. Perizinan akses ini dilakukan dengan menuliskannya ke dalam *file* *AndroidManifest.xml* yang terdapat dalam folder Android Studio Project. Contoh kode untuk menuliskan perizinan akses terhadap API dapat dilihat 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.

```
<uses-permission
    android:name="android.permission.PACKAGE_USAGE_STATS"
    tools:ignore="ProtectedPermissions" />
```

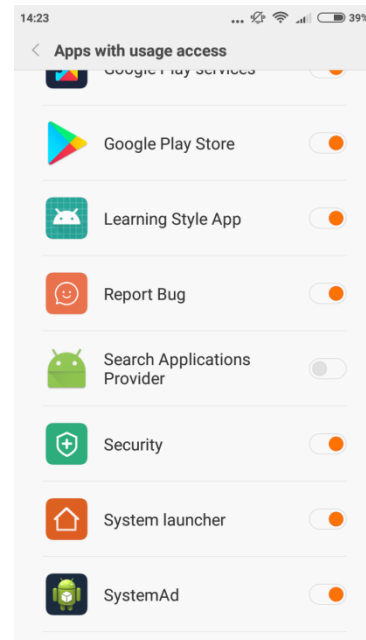
Gambar 4.2. Contoh kode untuk memberikan izin akses API

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).



a)



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 berbeda. 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 ketentuan standar waktu Android. Sedangkan untuk parameter waktu selesai diatur seakhir mungkin, yaitu waktu sekarang (ketika baris program `System.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()`.

```
List<UsageStats> usageStats = usageStatsManager.  
    queryUsageStats(UsageStatsManager.INTERVAL_YEARLY, beginTime, endTime);  
|  
for(UsageStats singleUsageStat : usageStats){  
    String packageName = singleUsageStat.getPackageName();  
    Long duration = singleUsageStat.getTotalTimeInForeground();  
    MyUsageStats myUsageStats = new MyUsageStats(packageName, duration);  
    myUsageStatsList.put(packageName, myUsageStats);  
}
```

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

```
Calendar calendar = Calendar.getInstance();  
calendar.set(Calendar.YEAR, 0);  
Long beginTime = calendar.getTimeInMillis();  
Long endTime = System.currentTimeMillis();
```

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_FOREGROUND saja 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.

```
UsageEvents usageEvents = usageStatsManager.queryEvents(beginTime, endTime);
while(usageEvents.hasNextEvent()){
    UsageEvents.Event event = new UsageEvents.Event();
    usageEvents.getNextEvent(event);
    if(event.getEventType() == UsageEvents.Event.MOVE_TO_FOREGROUND){
        MyUsageStats myUsageStats = myUsageStatsList.get(event.getPackageName());
        myUsageStats.frequency += 1;
    }
}
```

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

```

public class MyJob extends Job {
    public static void runJobImmediately() {
        int jobId = new JobRequest.Builder(MyJob.TAG)
            .startNow()
            .build()
            .schedule();
    }
}

```

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 merekam 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*

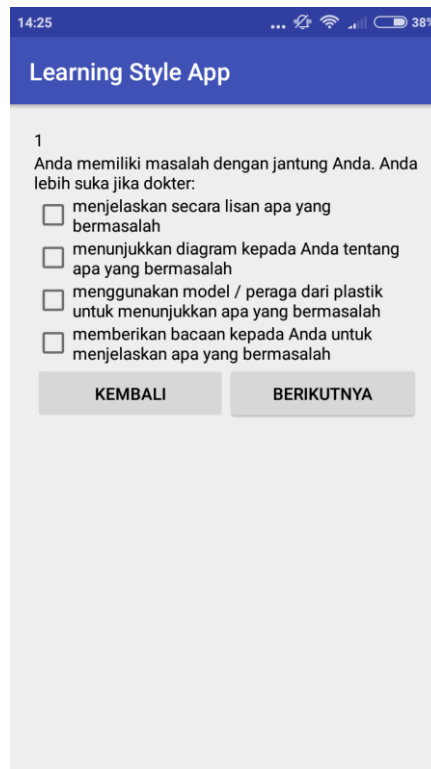
Prioritas	Product Backlog	Keterangan
1	Layanan Penangkap Data Penggunaan Aplikasi	
	Membuat menu akses pintas ke pengaturan perangkat bagian <i>App Usage Access</i>	Selesai
	Mengembangkan layanan penangkap data durasi penggunaan aplikasi	Selesai
	Mengembangkan layanan penangkap data frekuensi penggunaan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

4.1.2. *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.

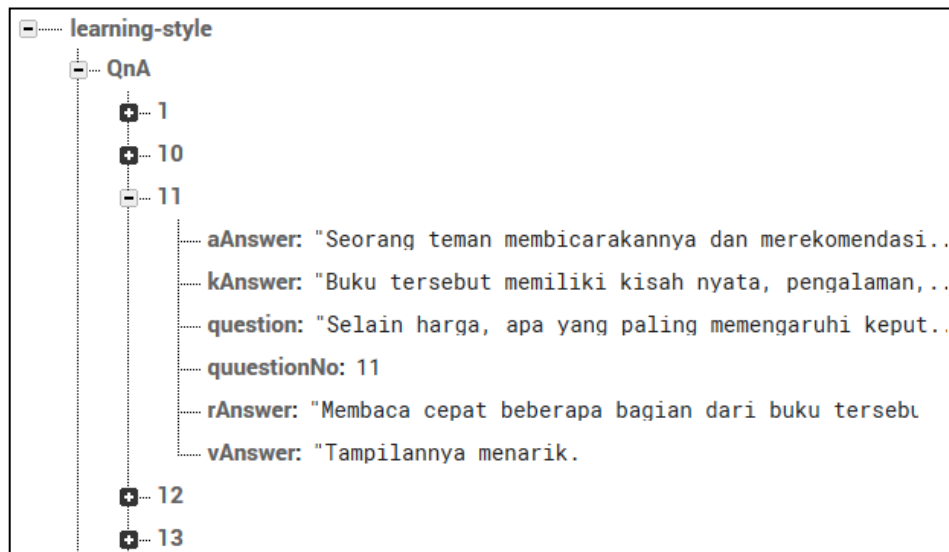
Tampilan halaman pengisian kuesioner VARK dapat dilihat seperti pada Gambar 4.9. Bagian atas halaman berisi pertanyaan dan nomor pertanyaan. Tersedia empat pilihan jawaban dalam bentuk *checkbox* sehingga responden dapat

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 *kAnswer*). 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

```
public class QuestionAndAnswers {
    public int questionNo;
    public String question;
    public String vAnswer, aAnswer, rAnswer, kAnswer;
    public int vPoint, aPoint, rPoint, kPoint;
    public ArrayList<String> answers = new ArrayList<String>();

    public QuestionAndAnswers(){}
}
```

Gambar 4.11. POJO kelas 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.

```

    public ArrayList<String> getAnswers() {
        if(answers.isEmpty()) initAnswers();
        return (answers);
    }
    public void initAnswers(){
        answers.add(vAnswer);
        answers.add(aAnswer);
        answers.add(rAnswer);
        answers.add(kAnswer);
        Collections.shuffle(answers);
    }
}

```

Gambar 4.12. Contoh kode program untuk mengacak urutan pilihan jawaban

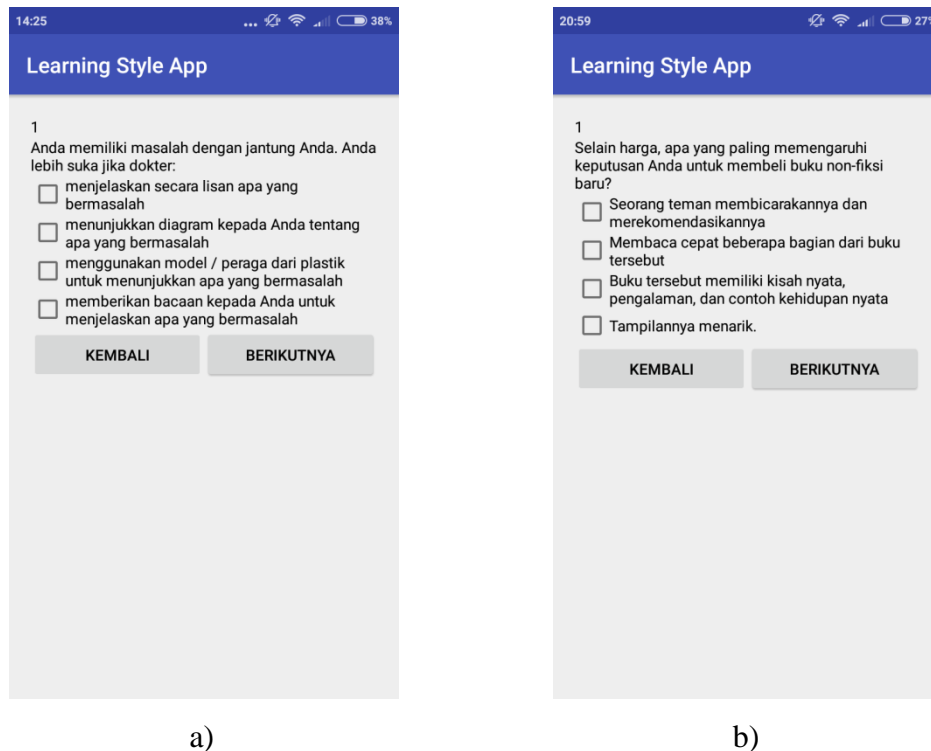
Setelah POJO dibuat, maka data dari Firebase dapat diambil dan ditampilkan ke dalam aplikasi. Contoh kode untuk mengambil data pertanyaan dan jawaban dapat dilihat pada Gambar 4.13. Selain urutan pilihan jawaban yang teracak, urutan pertanyaan yang ditampilkan juga dalam kondisi acak. Untuk itulah perintah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

```

mDatabase = FirebaseDatabase.getInstance();
mReference = mDatabase.getReference("QnA");
mReference.addListenerForSingleValueEvent(new ValueEventListener() {
    @Override
    public void onDataChange(DataSnapshot dataSnapshot) {
        for(DataSnapshot singleSnapshot : dataSnapshot.getChildren()){
            QuestionAndAnswers mQna = singleSnapshot.getValue(QuestionAndAnswers.class);
            mQna.initAnswers();
            qnaList.add(mQna);
        }
        Collections.shuffle(qnaList);
        startSurvey();
    }
});

```

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



a)

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.

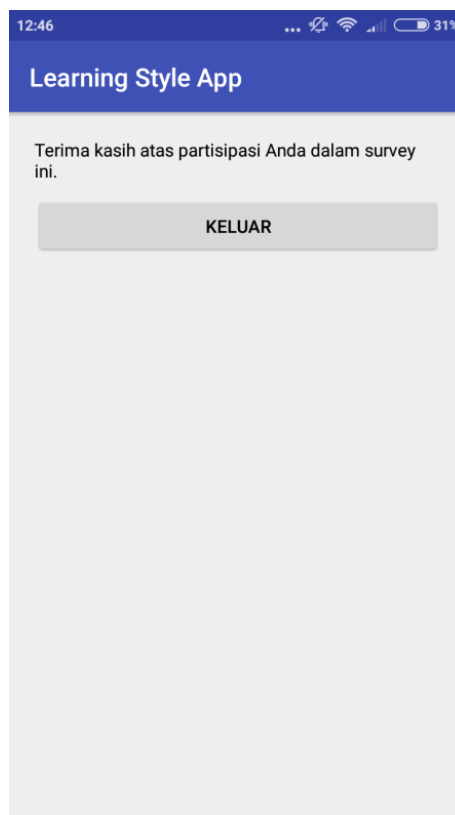
```
for(QuestionAndAnswers singleQna : qnaList){
    vTotal += singleQna.vPoint;
    aTotal += singleQna.aPoint;
    rTotal += singleQna.rPoint;
    kTotal += singleQna.kPoint;
}
```

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

```
Intent intent = new Intent(Intent.ACTION_MAIN);
intent.addCategory(Intent.CATEGORY_HOME);
intent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
startActivity(intent);
```

Gambar 4.18. Contoh kode program untuk keluar dari aplikasi

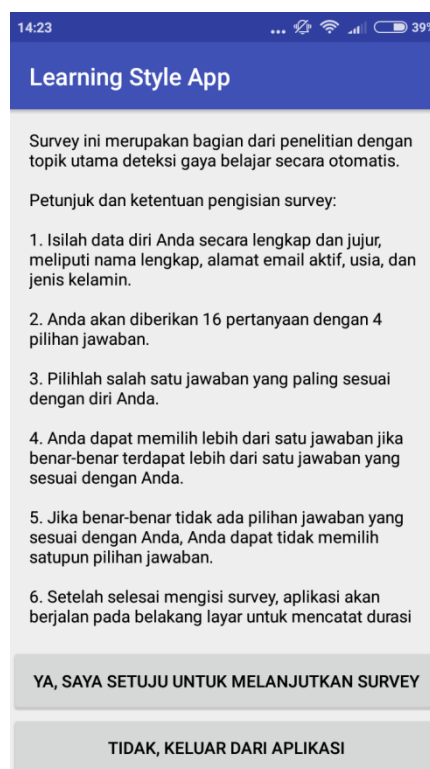
Setelah menyelesaikan fungsi-fungsi tersebut, maka *Sprint 2* telah selesai dilaksanakan. Daftar pertanyaan dan jawaban suervey VARK sudah tersimpan di dalam Firebase. Aplikasi sudah dapat mengambil data pertanyaan dan jawaban dari Firebase. Kemudian, aplikasi juga dapat menampilkan pertanyaan dan jawaban 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

4.1.3. *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 survey

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.

```
buttonTidak.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        finish();
    }
});
```

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.

14:24 39%

Learning Style App

Nama Lengkap
Meizar Raka Rimadana

Alamat email
meizar.rimadana@gmail.com

Usia
22

Jenis Kelamin
☒ Laki-laki ☐ Perempuan

MULAI SURVEY

Gambar 4.21. Halaman pengisian identitas responden

14:24

Learning Style App

Nama Lengkap

Alamat email
meizar.rimadana@gmail.com

Usia
22

Jenis Kelamin
☒ Laki-laki ☐ Perempuan

MULAI SURVEY

Harap masukkan nama Anda

a)

14:24

Learning Style App

Nama Lengkap
Meizar Raka Rimadana

Alamat email

Usia
22

Jenis Kelamin
☒ Laki-laki ☐ Perempuan

MULAI SURVEY

Harap masukkan alamat email Anda

b)

14:25

Learning Style App

Nama Lengkap
Meizar Raka Rimadana

Alamat email
meizar.rimadana@gmail.com

Usia

Jenis Kelamin
☒ Laki-laki ☐ Perempuan

MULAI SURVEY

Harap masukkan usia Anda

c)

14:25

Learning Style App

Nama Lengkap
Meizar Raka Rimadana

Alamat email
meizar.rimadana@gmail.com

Usia
22

Jenis Kelamin
☐ Laki-laki ☒ Perempuan

MULAI SURVEY

Harap pilih jenis kelamin Anda

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.

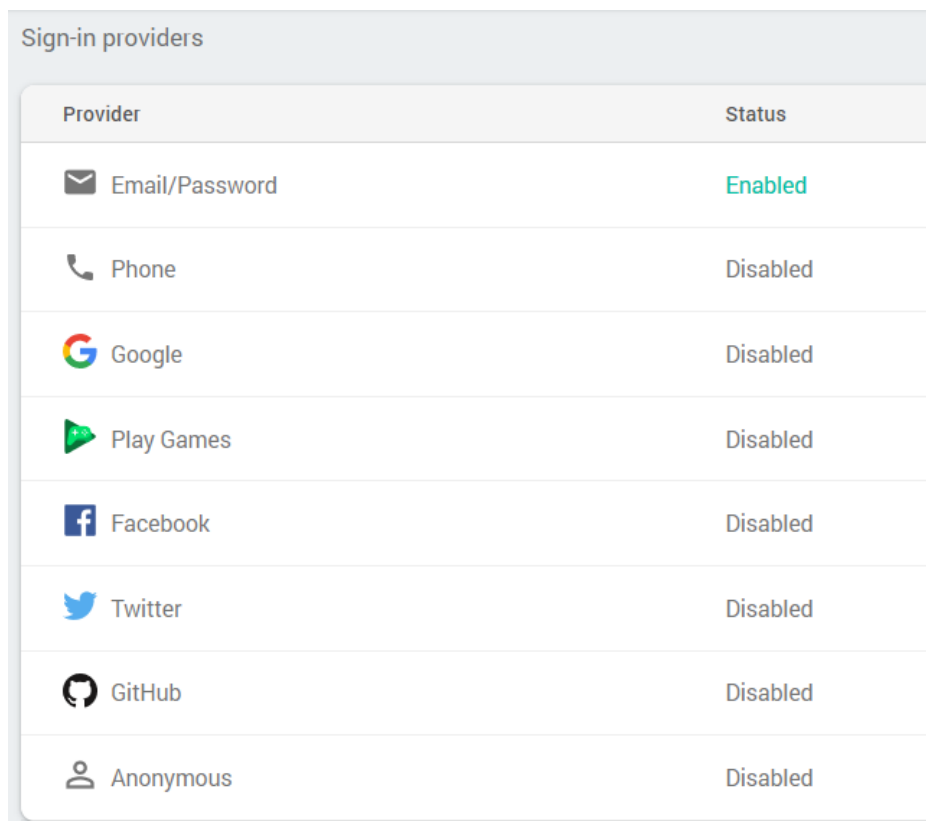
Tabel 4.4. *Sprint 3 Overview*









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

4.1.4. *Sprint 4 : Sinkronisasi Firebase, Autentikasi, dan Penyelesaian Akhir*

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.



Sign-in providers	
Provider	Status
 Email/Password	Enabled
 Phone	Disabled
 Google	Disabled
 Play Games	Disabled
 Facebook	Disabled
 Twitter	Disabled
 GitHub	Disabled
 Anonymous	Disabled

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 mengisi 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.

```
firebaseAuth.createUserWithEmailAndPassword(email, password)
    .addOnCompleteListener(StartActivity.this, new OnCompleteListener<AuthResult>() {
        @Override
        public void onComplete(@NonNull Task<AuthResult> task) {
            progressBar.setVisibility(View.GONE);
            if (!task.isSuccessful()) {
                Toast.makeText(StartActivity.this, task.getException()
                    .getMessage(), Toast.LENGTH_SHORT).show();
            }
            else {
                startActivity(new Intent(StartActivity.this, SurveyActivity.class));
            }
        }
    });
```

Gambar 4.24. Contoh kode program 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 sebuah kelas 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.

```

public class User {
    public String uid;
    public String nama;
    public String email;
    public int usia;
    public int gender;

    public User(){}
    public User(String uid, String nama, String email, int usia, int gender){
        this.uid = uid;
        this.nama = nama;
        this.email = email;
        this.usia = usia;
        this.gender = gender;
    }
}

```

Gambar 4.25.POJO kelas User

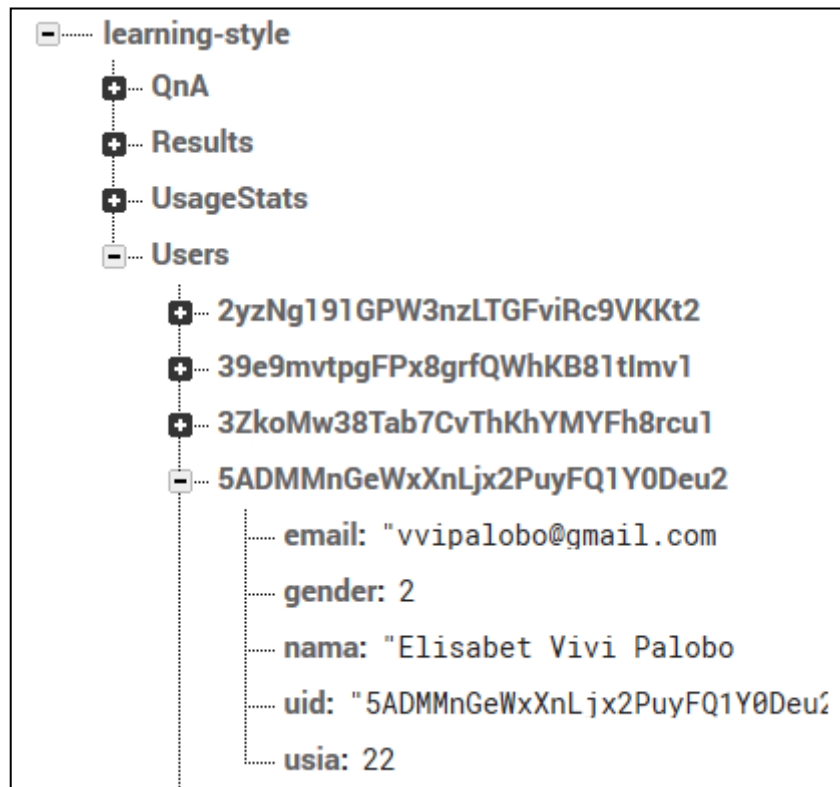
Kemudian obyek User akan disimpan ke basis data Firebase dengan node “Users” berdasarkan *uid* dari pengguna yang bersangkutan seperti pada Gambar 4.26. Firebase secara otomatis akan memberikan *uid* yang berbeda untuk setiap pengguna. Untuk mendapatkan data *uid* tersebut, firebase menyediakan fungsi `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.

```

FirebaseUser firebaseUser = FirebaseAuth.getInstance()
    .getCurrentUser();
String uid = firebaseUser.getUid();
User user = new User(uid, nama, email, usia, gender);
mDatabase.child("Users").child(uid).setValue(user);

```

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 pengguna, 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.

```

public class Result {
    public String uid;
    public int v;
    public int a;
    public int r;
    public int k;

    public Result(){}
    public Result(String uid, int v, int a, int r, int k){
        this.uid = uid;
        this.v = v;
        this.a = a;
        this.r = r;
        this.k = k;
    }
}

```

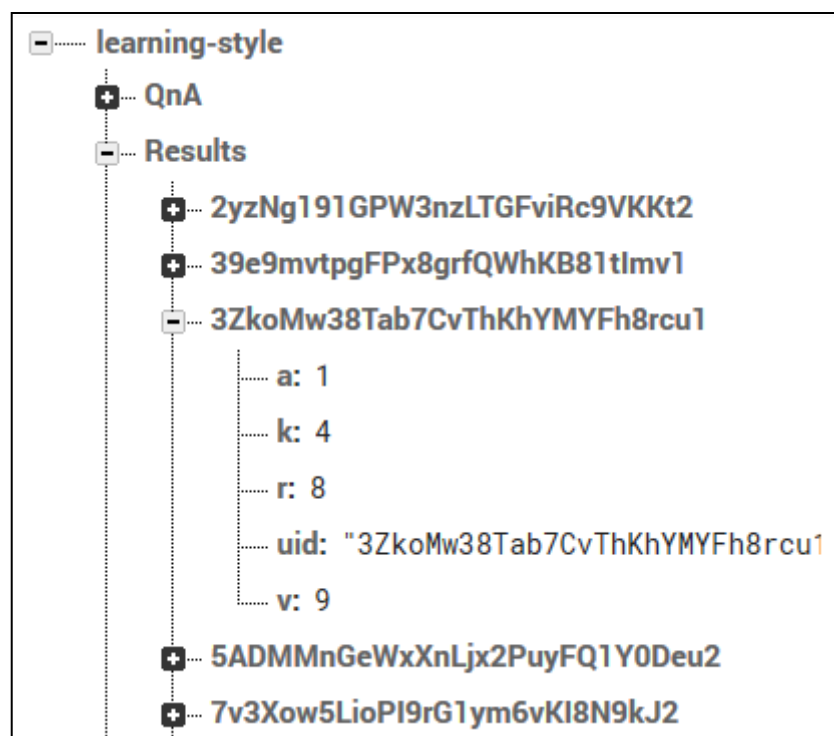
Gambar 4.28.Kelas POJO Result

```

Result result = new Result(firebaseUser.getId(), vTotal, aTotal, rTotal, kTotal);
resultReference.child(result.uid).setValue(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*).

```
public class MyUsageStats {
    public String packageName;
    public int frequency;
    public long duration;

    public MyUsageStats(){}
    public MyUsageStats(String packageName, long duration){
        this.packageName = packageName;
        this.duration = duration;
    }

    public void setFrequency(int frequency){
        this.frequency = frequency;
    }
}
```

Gambar 4.31. POJO kelas `MyUsageStats`

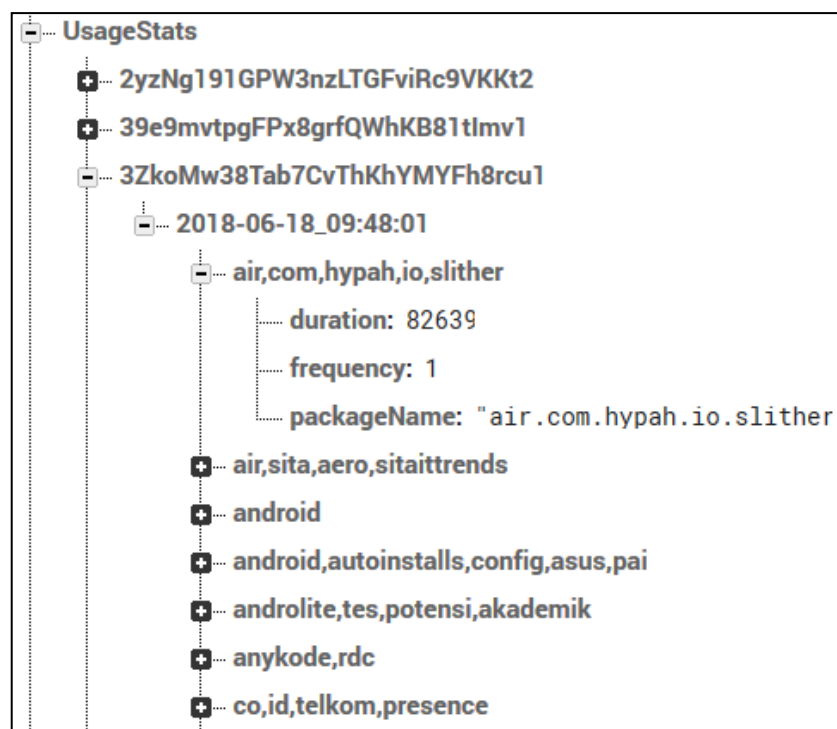
Berikutnya, obyek tersebut akan disimpan ke dalam Firebase dalam node `UsageStats` berdasarkan *Unique ID* (UID) pengguna tersebut seperti pada Gambar 4.32. Data dilengkapi dengan penanda waktu (*timestamp*) terlebih dahulu untuk mengetahui kapan data tersebut diambil. Hal ini bertujuan untuk mengantisipasi 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.

```

DatabaseReference mReference = FirebaseDatabase.getInstance()
    .getReference("UsageStats");
FirebaseUser firebaseUser = FirebaseAuth.getInstance().getCurrentUser();
Date now = new Date(endTime);
String nowString = SimpleDateFormat.format(now);
for(MyUsageStats result : myUsageStatsList.values()){
    String key = result.packageName.replace(".", ",");
    mReference.child(firebaseUser.getUid()).child(nowString).child(key)
        .setValue(result);
}

```

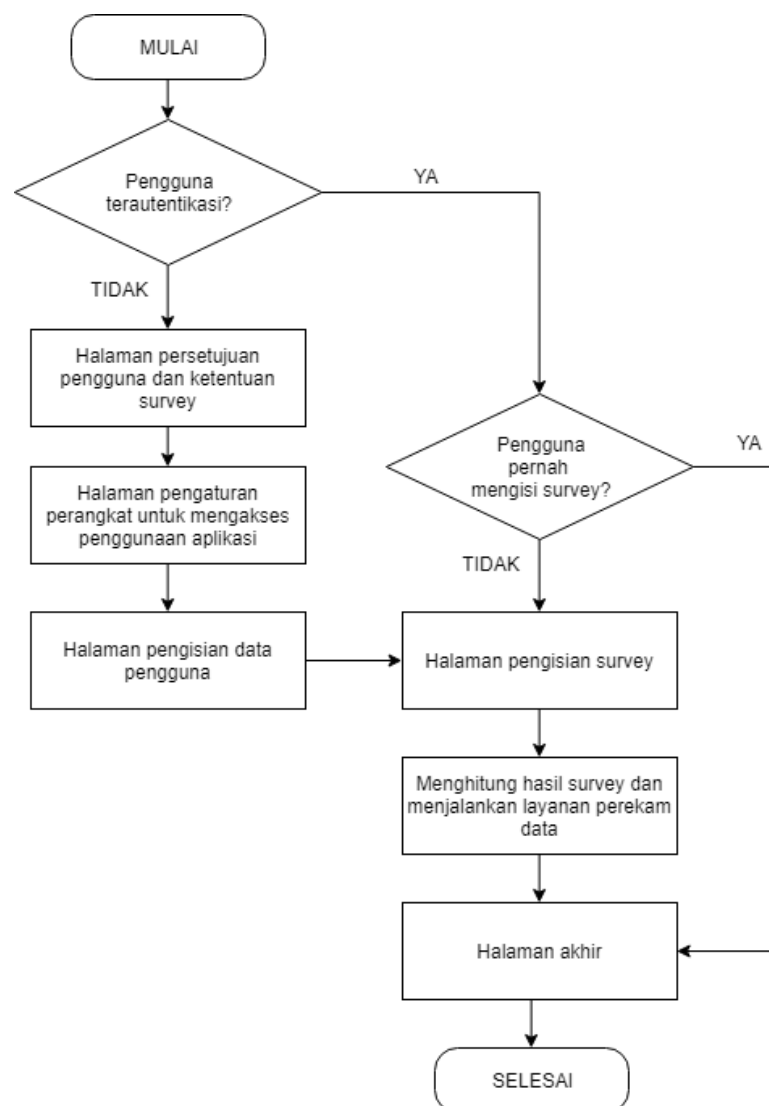
Gambar 4.32. Contoh kode program untuk menyimpan data ke Firebase



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


```

FirebaseAuth firebaseAuth = FirebaseAuth.getInstance();
if(firebaseAuth.getCurrentUser() != null){
    Intent intent = new Intent(PetunjukActivity.this, SurveyActivity.class);
    startActivity(intent);
    finish();
}

```

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

```

firebaseUser = FirebaseAuth.getInstance().getCurrentUser();
resultReference = FirebaseDatabase.getInstance().getReference("Results");
resultReference.addListenerForSingleValueEvent(new ValueEventListener() {
    @Override
    public void onDataChange(DataSnapshot dataSnapshot) {
        if(dataSnapshot.hasChild(firebaseUser.getUid())){
            Intent intent = new Intent(SurveyActivity.this, ResultActivity.class);
            startActivity(intent);
        }
    }

    @Override
    public void onCancelled(DatabaseError databaseError) {

    }
});

```

Gambar 4.36. Contoh kode program untuk memeriksa apakah pengguna sudah pernah 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 survey 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

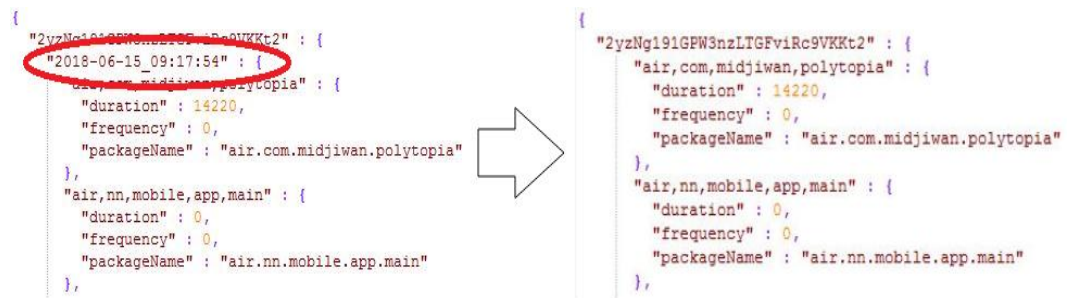
yang dilakukan tersebut sesuai dengan *product backlog* yang telah disusun sebelumnya seperti pada Tabel 4.5.

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

4.2. Analisis Uji Korelasi

Data yang akan diambil dari Firebase dan digunakan dalam analisis adalah “Results” dan “UsageStats”. Data “Results” berisikan nilai VARK aktual dari tiap responden. Data “UsageStats” berisikan hasil perekaman data penggunaan aplikasi *smartphone* dari tiap pengguna. Pada “UsageStats”, data hasil rekaman terkandung dalam sebuah *timestamp* dari waktu pengambilan data. Data tersebut tidak signifikan pada analisis yang akan dilakukan, sehingga dilakukan perubahan struktur data terlebih dahulu. Perubahan struktur tersebut dilakukan dengan menghilangkan *timestamp* seperti yang ditampilkan pada Gambar 4.37. Gambar tersebut menampilkan perubahan yang terjadi pada suatu penggalan data.



Gambar 4.37. Ilustrasi Perubahan Struktur Data

Hasil perekaman data pada Firebase hanya tersedia dalam format *JavaScript Object Notation* (JSON), sedangkan format data yang lebih efisien untuk diolah menggunakan Python adalah comma-separated values (CSV). Sehingga, perlu dilakukan konversi data terlebih dahulu. Konversi data dilakukan menggunakan layanan konversi daring <https://json-csv.com/>. Layanan tersebut hanya mampu mengolah data dengan ukuran maksimal 1 MB, sedangkan data “UsageStats” memiliki ukuran 1,206 MB. Sehingga, data tersebut perlu dibagi 2 terlebih dahulu sebelum dilakukan konversi. Proses ini menghasilkan 3 tabel: tabel nilai VARK aktual (“actualVark”), tabel rekaman aplikasi 1 (“usageStats1”), dan tabel rekaman aplikasi 2 (“usageStats2”). Adapun penggalan struktur tabel dari data yang telah dikonversi ditampilkan pada Tabel 4.6 dan Tabel 4.7.

Tabel 4.6. Tabel Nilai VARK Aktual

uid	v	a	r	k
2yzNg191GPW3nzLTGFviRc9VKkt2	9	7	10	11
39e9mvtpgFPx8grfQWhKB81tlmv1	8	9	8	4
3ZkoMw38Tab7CvThKhYMYFh8rcu1	9	1	8	4
5ADMMnGeWxXnLjx2PuyFQ1Y0Deu2	1	6	10	5
7v3Xow5LioPI9rG1ym6vKI8N9kJ2	8	2	9	5
D4Nvi0aWY7ZvRaxlkrWwk9XDftA3	2	8	3	13
DUwqt99Br9QkLepIeDNyA75snZe2	5	6	5	5

Tabel 4.7. Tabel Data Penggunaan Aplikasi

uid	package_name	duration	frequency
2yzNg191GPW3nzLTGFviRc9VKKt2	air.com.midjiwan.polytopia	14220	0
	air.nn.mobile.app.main	0	0
	all.in.one.calculator	0	0
	android	14212	10
	asd.vector.sensor	31198	0
	bpom.webreg	484007	0
	cc.forestapp	239229	0
	cn.wps.moffice_eng	59519	0

Data yang telah dikonversi akan diolah lebih lanjut. Terdapat 2 tahapan utama dalam analisis data yang akan dilakukan, yaitu pengolahan awal data dan uji korelasi. Analisis uji korelasi dilakukan dengan bantuan perangkat lunak SPSS. Analisis dilakukan antara nilai ukuran pola pengamatan dari penggunaan suatu aplikasi atau kelompok aplikasi, terhadap tiap kelas dari VARK dengan model Spearman. Model Spearman digunakan karena data tidak terdistribusi normal dan tidak terdapat hubungan linear antar peubah, sehingga tidak layak untuk diterapkan pada model Pearson. Batas tingkat signifikansi yang digunakan adalah 0,05. Korelasi akan dinyatakan signifikan ketika tingkat signifikansinya berada di bawah 0,05. Analisis uji korelasi akan dilakukan dalam 3 skenario yang berbeda. Titik perbedaan pada tiap skenario terdapat pada bagian pengelompokan aplikasi. Skenario pertama tidak melalui pengelompokan aplikasi. Skenario kedua mengelompokkan aplikasi ke dalam kelas-kelas VARK berdasarkan kelas yang paling dominan dari responden. Skenario ketiga mengelompokkan aplikasi berdasarkan fungsinya.

4.2.1. Pengolahan Awal Data

Pengolahan awal dilakukan untuk memperbaiki struktur data supaya lebih optimal untuk digunakan dalam pengujian. Pengolahan data dilakukan menggunakan Python. Python digunakan karena bahasa pemrograman ini kaya akan pustaka-pustaka pemrograman yang dapat digunakan untuk pengolahan dan analisis data. Selain itu, pengolahan data menggunakan Python juga lebih efisien dan fleksibel. Sebagai contoh kasus perbandingan, untuk menerapkan suatu fungsi komputasi yang cukup rumit menggunakan Excel, dibutuhkan pendefinisian

sejumlah kolom atau bahkan *sheet* baru. Sedangkan fungsi komputasi yang sama dapat diterapkan menggunakan Python hanya dengan 1 baris kode saja. Python lebih fleksibel karena tiap informasi pada tabel dapat disimpan sebagai irisan-irisan independen dalam peubah yang dapat diolah dan ditampilkan dengan mudah. Pustaka pemrograman yang digunakan dalam pengolahan data adalah *pandas*. Pustaka *pandas* merupakan pustaka yang dikembangkan untuk melakukan pengolahan dan analisis data dengan Python. Tahap pertama pengolahan data adalah penggabungan kembali dan pembersihan data pada tabel rekaman penggunaan aplikasi. Adapun kode pemrograman yang diterapkan untuk tahapan ini beserta penjelasan singkatnya ditampilkan pada Gambar 4.38.

```
import pandas as pd #pustaka analisis data di Python

#1. Penggabungan dan pembersihan data rekaman penggunaan aplikasi

usage1 = pd.read_csv('_0_usageStats1.csv')
usage2 = pd.read_csv('_0_usageStats2.csv')

#Penggabungan 2 data penggunaan aplikasi yang terpisah
usageStats = pd.concat([usage1,usage2])

#Menghapus semua baris yang memiliki nilai 0 pada kedua kolom: durasi dan frekuensi
usageStats = usageStats[(usageStats[['duration','frequency']] != 0).any(axis=1)]

#Melakukan pengisian "uid" yang kosong
usageStats.uid.fill(inplace=True)
```

Gambar 4.38. Kode pemrograman untuk pengolahan data tahap pertama

Pertama-tama perlu dilakukan penyatuan kembali terhadap data rekaman aplikasi yang sempat dipisahkan. Penggabungan data dilakukan dengan fungsi *concatenation* dari *pandas*. Fungsi ini secara *default* menggabungkan 2 dua kerangka data secara bertingkat. Setelah itu, dilakukan pembersihan data untuk menghapus semua baris yang memiliki nilai 0 (tidak bernilai) pada kolom durasi dan frekuensinya secara bersama-sama. Proses ini dilakukan dengan mendefinisikan suatu pernyataan kondisional yang kemudian akan divalidasi dengan fungsi *any*. Fungsi tersebut mengembalikan nilai *true* ketika salah satu syarat terpenuhi, dan *false* ketika semua syarat tidak terpenuhi. Fungsi ini digunakan supaya baris yang memiliki nilai 0 hanya pada salah satu tabel masih dipertahankan dan tidak ikut terhapus, karena masih layak digunakan pada

pengujian. Tabel rekaman penggunaan aplikasi ini juga membutuhkan pelengkapan data. Seperti yang ditampilkan pada Tabel 4.x, data id responden (uid) hanya terisi pada baris paling atas saja untuk tiap kelompok aplikasi dari tiap responden. Fungsi yang digunakan untuk melengkapi data id responden adalah *forward fill*. Fungsi tersebut secara otomatis mengisi sel-sel yang masih kosong dengan nilai valid pertama yang ditemuinya, hingga menemukan nilai valid selanjutnya. Hasil pengolahan awal data ditampilkan pada Tabel 4.8. Data rekaman penggunaan aplikasi yang telah terolah akan memasuki tahap selanjutnya, yaitu pengelompokan aplikasi. Mulai dari tahap ini, alur analisis data sudah berbeda antara ketiga skenario uji korelasi.

Tabel 4.8. Hasil Pengolahan Awal terhadap Data Penggunaan Aplikasi

uid	package_name	duration	frequency
2yzNg191GPW3nzLTGFviRc9VKKt2	air.com.midjiwan.polytopia	14220	0
2yzNg191GPW3nzLTGFviRc9VKKt2	android	14212	10
2yzNg191GPW3nzLTGFviRc9VKKt2	asd.vector.sensor	31198	0
2yzNg191GPW3nzLTGFviRc9VKKt2	bpom.webreg	484007	0
2yzNg191GPW3nzLTGFviRc9VKKt2	cc.forestapp	239229	0
2yzNg191GPW3nzLTGFviRc9VKKt2	cn.wps.moffice_eng	59519	0
2yzNg191GPW3nzLTGFviRc9VKKt2	com.Akaelenka.Flica	2013269	0
2yzNg191GPW3nzLTGFviRc9VKKt2	com.GamesFromGarage.PCArchitect	9812800	0

4.2.2. Analisis Uji Korelasi Skenario 1

Pada skenario ini, aplikasi diuji secara bebas tanpa melalui proses pengelompokan. Akan tetapi, masih perlu dilakukan proses pengolahan yaitu melakukan penyaringan dan normalisasi data. Proses tersebut diterapkan melalui kode pemrograman yang ditampilkan pada Gambar 4.39.

```

#Menghapus aplikasi yang tidak signifikan
df = df[~(df.package_name.isin(excluded))].reset_index(drop=True)

#Menuliskan hasil ke dalam format .xlsx
writer = pd.ExcelWriter('Dataset2.xlsx')

#Daftar pola pengamatan
patterns = ['duration', 'frequency']

#Fungsi pembentukan dataset uji korelasi
def build_testDf(pola):
    #Menghapus nilai 0 pada kolom pola yang dipilih
    filtered = df.loc[(df[pola]!=0), ['uid', 'package_name', pola]]
    #Menyaring data dari data-data dengan jumlah sample kurang dari 5
    filtered = filtered.groupby('package_name').filter(lambda x: len(x) > 4)
    #Menerapkan fungsi normalisasi pada kolom pola untuk tiap pengguna (uid)
    filtered[pola] = filtered.groupby('uid')[pola].apply(lambda x: (x-min(x))/(max(x)-min(x)))
    #Menggabungkan data ukuran pola dengan nilai VARK aktual
    dataset = pd.merge(filtered, df2, on='uid')
    #Menuliskan data ke dalam sheet excel
    dataset.to_excel(writer, pola, index=False)

```

Gambar 4.39. Kode Pemrograman untuk Pengolahan Data pada Skenario 1

Proses penyaringan dilakukan dalam 2 tahap. Tahap pertama penyaringan dilakukan terhadap data dari aplikasi-aplikasi yang tidak signifikan terhadap VARK, seperti aplikasi konfigurasi sistem, alarm, *bluetooth*, dll. Daftar aplikasi yang akan dihapus didefinisikan dalam suatu peubah, kemudian dilakukan penyaringan terhadap kolom “package_name” berdasarkan peubah tersebut. Tahap kedua penyaringan dilakukan untuk menghapus aplikasi-aplikasi yang muncul kurang dari 5 kali. Hal ini dilakukan untuk membatasi jumlah minimum dari sampel pengujian. Selanjutnya merupakan proses pembentukan *dataset* pengujian. Proses ini dilakukan secara iteratif untuk tiap pola pengamatan, dan akan disimpan ke dalam *sheet* dari suatu *file* Excel. Dalam proses ini, diterapkan fungsi normalisasi pada nilai-nilai pola pengamatan yang dikelompokkan berdasarkan responden. Normalisasi perlu dilakukan karena tiap responden memiliki rentang nilai pola yang berbeda-beda. Normalisasi berfungsi untuk menyetarakan rentang nilai pola bagi semua responden menjadi [0, 1]. Tahap terakhir merupakan penggabungan hasil normalisasi data rekaman penggunaan aplikasi dengan data nilai VARK aktual menjadi suatu *dataset* yang akan digunakan dalam analisis uji korelasi. Adapun gambaran struktur data dari hasil pembentukan *dataset* ini ditampilkan pada Tabel 4.9.

Tabel 4.9. Tabel *Dataset* Skenario 1

uid	package_name	duration	v	a	r	k
2yzNg191GPW3nzLTGFvIRc9VKKt2	cn.wps.moffice_eng	0.000228	9	7	10	11
2yzNg191GPW3nzLTGFvIRc9VKKt2	com.UCMobile.intl	1	9	7	10	11
2yzNg191GPW3nzLTGFvIRc9VKKt2	com.android.browser	0.000304	9	7	10	11
2yzNg191GPW3nzLTGFvIRc9VKKt2	com.android.camera	0.000282	9	7	10	11
2yzNg191GPW3nzLTGFvIRc9VKKt2	com.android.chrome	0.028738	9	7	10	11
2yzNg191GPW3nzLTGFvIRc9VKKt2	com.android.htmlviewer	0.000327	9	7	10	11
2yzNg191GPW3nzLTGFvIRc9VKKt2	com.android.mms	0.00025	9	7	10	11
2yzNg191GPW3nzLTGFvIRc9VKKt2	com.android.phone	0.000425	9	7	10	11
2yzNg191GPW3nzLTGFvIRc9VKKt2	com.android.thememanager	3.97E-06	9	7	10	11
2yzNg191GPW3nzLTGFvIRc9VKKt2	com.bitsmedia.android.muslimpro	0.000886	9	7	10	11

Hasil pengolahan data pengujian yang terbentuk dimasukkan ke dalam perangkat lunak SPSS untuk analisis uji korelasi. Uji korelasi menunjukkan adanya sejumlah hubungan yang signifikan antara penggunaan aplikasi pada *smartphone* responden dengan nilai VARK aktual responden yang akan dijabarkan sebagai berikut.

Tabel 4.10. Hasil Pengujian Durasi Penggunaan “com.adobe.lrmobile”

Correlations ^a			duration	v	a	r	k
Spearman's rho	duration	Correlation Coefficient	1.000	.205	.900 [*]	.300	.821
		Sig. (2-tailed)	.	.741	.037	.624	.089
		N	5	5	5	5	5
	v	Correlation Coefficient	.205	1.000	-.154	-.872	.526
		Sig. (2-tailed)	.741	.	.805	.054	.362
		N	5	5	5	5	5
	a	Correlation Coefficient	.900 [*]	-.154	1.000	.600	.718
		Sig. (2-tailed)	.037	.805	.	.285	.172
		N	5	5	5	5	5
	r	Correlation Coefficient	.300	-.872	.600	1.000	-.103
		Sig. (2-tailed)	.624	.054	.285	.	.870
		N	5	5	5	5	5
	k	Correlation Coefficient	.821	.526	.718	-.103	1.000
		Sig. (2-tailed)	.089	.362	.172	.870	.
		N	5	5	5	5	5

*. Correlation is significant at the 0.05 level (2-tailed).

a. package_name = com.adobe.lrmobile

Tabel 4.10 menampilkan hasil uji korelasi antara durasi penggunaan aplikasi “com.adobe.lrmobile” terhadap VARK. Aplikasi ini merupakan suatu aplikasi pengambil, penyunting, dan pembagi gambar. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,037. Nilai tersebut lebih kecil dari

batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi “com.adobe.lrmobile” memiliki korelasi yang signifikan terhadap kelas A. Koefisien korelasi memiliki nilai 0.9 yang berarti kedua peubah ini memiliki korelasi yang bersifat sangat kuat dan positif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi “com.adobe.lrmobile” oleh seorang responden, maka semakin tinggi kecenderungan preferensi responden tersebut terhadap informasi yang bersifat aural. Hasil berikutnya adalah sebagai berikut.

Tabel 4.11. Hasil Pengujian Durasi Penggunaan “com.android.calculator2”

Correlations ^a			duration	v	a	r	k
Spearman's rho	duration	Correlation Coefficient	1.000	-.255	-.324	-.533*	-.201
		Sig. (2-tailed)	.	.360	.239	.041	.472
		N	15	15	15	15	15
	v	Correlation Coefficient	-.255	1.000	.655**	.598*	.556*
		Sig. (2-tailed)	.360	.	.008	.019	.031
		N	15	15	15	15	15
	a	Correlation Coefficient	-.324	.655**	1.000	.478	.771**
		Sig. (2-tailed)	.239	.008	.	.072	.001
		N	15	15	15	15	15
	r	Correlation Coefficient	-.533*	.598*	.478	1.000	.121
		Sig. (2-tailed)	.041	.019	.072	.	.667
		N	15	15	15	15	15
	k	Correlation Coefficient	-.201	.556*	.771**	.121	1.000
		Sig. (2-tailed)	.472	.031	.001	.667	.
		N	15	15	15	15	15

*. Correlation is significant at the 0.05 level (2-tailed).

**. Correlation is significant at the 0.01 level (2-tailed).

a. package_name = com.android.calculator2

Tabel 4.11 menampilkan hasil uji korelasi antara durasi penggunaan aplikasi “com.android.calculator2” terhadap VARK. Aplikasi ini merupakan salah satu aplikasi kalkulator pada *smartphone* android. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,041. Nilai tersebut lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi “com.android.calculator2” memiliki korelasi yang signifikan terhadap kelas R. Koefisien korelasi memiliki nilai -0,533 yang berarti kedua peubah ini memiliki korelasi yang bersifat lumayan kuat dan negatif. Hal

ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi “com.android.calculator2” oleh seorang responden, maka semakin rendah kecenderungan preferensi responden tersebut terhadap informasi yang bersifat tekstual terstruktur. Hasil berikutnya adalah sebagai berikut.

Tabel 4.12. Hasil Pengujian Durasi Penggunaan “com.android.mms”

Correlations ^a			duration	v	a	r	k
Spearman's rho	duration	Correlation Coefficient	1.000	-.139	-.399*	-.172	-.240
		Sig. (2-tailed)	.	.507	.048	.411	.249
		N	25	25	25	25	25
	v	Correlation Coefficient	-.139	1.000	.449*	.472*	.447*
		Sig. (2-tailed)	.507	.	.024	.017	.025
		N	25	25	25	25	25
	a	Correlation Coefficient	-.399*	.449*	1.000	.276	.534**
		Sig. (2-tailed)	.048	.024	.	.182	.006
		N	25	25	25	25	25
	r	Correlation Coefficient	-.172	.472*	.276	1.000	.370
		Sig. (2-tailed)	.411	.017	.182	.	.069
		N	25	25	25	25	25
	k	Correlation Coefficient	-.240	.447*	.534**	.370	1.000
		Sig. (2-tailed)	.249	.025	.006	.069	.
		N	25	25	25	25	25

*. Correlation is significant at the 0.05 level (2-tailed).

**. Correlation is significant at the 0.01 level (2-tailed).

a. package_name = com.android.mms

Tabel 4.12 menampilkan hasil uji korelasi antara durasi penggunaan aplikasi “com.android.mms” terhadap VARK. Aplikasi ini merupakan aplikasi pengirim pesan multimedia pada *smartphone* android. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,048. Nilai tersebut lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi “com.android.mms” memiliki korelasi yang signifikan terhadap kelas A. Koefisien korelasi memiliki nilai -0,399 yang berarti kedua peubah ini memiliki korelasi yang bersifat lemah dan negatif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi “com.android.mms” oleh seorang responden, maka semakin rendah kecenderungan preferensi responden tersebut terhadap informasi yang bersifat aural. Hasil berikutnya adalah sebagai berikut.

Tabel 4.13. Hasil Pengujian Durasi Penggunaan “com.duapps.recorder”

Correlations ^a			duration	v	a	r	k
Spearman's rho	duration	Correlation Coefficient	1.000	.410	.894*	-.359	.100
		Sig. (2-tailed)	.	.493	.041	.553	.873
		N	5	5	5	5	5
	v	Correlation Coefficient	.410	1.000	.459	.395	.872
		Sig. (2-tailed)	.493	.	.437	.511	.054
		N	5	5	5	5	5
	a	Correlation Coefficient	.894*	.459	1.000	.000	.224
		Sig. (2-tailed)	.041	.437	.	1.000	.718
		N	5	5	5	5	5
	r	Correlation Coefficient	-.359	.395	.000	1.000	.410
		Sig. (2-tailed)	.553	.511	1.000	.	.493
		N	5	5	5	5	5
	k	Correlation Coefficient	.100	.872	.224	.410	1.000
		Sig. (2-tailed)	.873	.054	.718	.493	.
		N	5	5	5	5	5

*. Correlation is significant at the 0.05 level (2-tailed).

a. package_name = com.duapps.recorder

Tabel 4.13 menampilkan hasil uji korelasi antara durasi penggunaan aplikasi “com.duapps.recorder” terhadap VARK. Aplikasi ini memiliki sejumlah fungsi seperti perekam layar, penyunting video, siaran langsung, *video call*, dll. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,041. Nilai tersebut lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi “com.duapps.recorder” memiliki korelasi yang signifikan terhadap kelas A. Koefisien korelasi memiliki nilai 0.894 yang berarti kedua peubah ini memiliki korelasi yang sangat kuat dan positif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi “com.duapps.recorder” oleh seorang responden, maka semakin tinggi kecenderungan preferensi responden tersebut terhadap informasi yang bersifat aural. Hasil berikutnya adalah sebagai berikut.

Tabel 4.14. Hasil Pengujian Durasi Penggunaan “com.microsoft.office.word”

Correlations ^a			duration	v	a	r	k
Spearman's rho	duration	Correlation Coefficient	1.000	.000	-.815**	-.159	-.136
		Sig. (2-tailed)	.	1.000	.007	.683	.728
		N	9	9	9	9	9
	v	Correlation Coefficient	.000	1.000	.262	.145	.654
		Sig. (2-tailed)	1.000	.	.496	.709	.056
		N	9	9	9	9	9
	a	Correlation Coefficient	-.815**	.262	1.000	-.034	.252
		Sig. (2-tailed)	.007	.496	.	.931	.513
		N	9	9	9	9	9
	r	Correlation Coefficient	-.159	.145	-.034	1.000	.340
		Sig. (2-tailed)	.683	.709	.931	.	.370
		N	9	9	9	9	9
	k	Correlation Coefficient	-.136	.654	.252	.340	1.000
		Sig. (2-tailed)	.728	.056	.513	.370	.
		N	9	9	9	9	9

** . Correlation is significant at the 0.01 level (2-tailed).

a. package_name = com.microsoft.office.word

Tabel 4.14 menampilkan hasil uji korelasi antara durasi penggunaan aplikasi “com.microsoft.office.word” terhadap VARK. Aplikasi ini merupakan aplikasi pengolah kata. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,007. Nilai ini lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi “com.microsoft.office.word” memiliki korelasi yang signifikan terhadap kelas A. Koefisien korelasi memiliki nilai -0.815 yang berarti kedua peubah ini memiliki korelasi yang sangat kuat dan negatif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi “com.microsoft.office.word” oleh seorang responden, maka semakin rendah kecenderungan preferensi responden tersebut terhadap informasi yang bersifat aural. Hasil berikutnya adalah sebagai berikut.

Tabel 4.15. Hasil Pengujian Durasi Penggunaan “com.mobile.legends”

Correlations ^a			duration	v	a	r	k
Spearman's rho	duration	Correlation Coefficient	1.000	-.667	-.943**	-.765	-.143
		Sig. (2-tailed)	.	.148	.005	.076	.787
		N	6	6	6	6	6
	v	Correlation Coefficient	-.667	1.000	.493	.940**	.319
		Sig. (2-tailed)	.148	.	.321	.005	.538
		N	6	6	6	6	6
	a	Correlation Coefficient	-.943**	.493	1.000	.618	.314
		Sig. (2-tailed)	.005	.321	.	.191	.544
		N	6	6	6	6	6
	r	Correlation Coefficient	-.765	.940**	.618	1.000	.265
		Sig. (2-tailed)	.076	.005	.191	.	.612
		N	6	6	6	6	6
	k	Correlation Coefficient	-.143	.319	.314	.265	1.000
		Sig. (2-tailed)	.787	.538	.544	.612	.
		N	6	6	6	6	6

** . Correlation is significant at the 0.01 level (2-tailed).

a. package_name = com.mobile.legends

Tabel 4.15 menampilkan hasil uji korelasi antara durasi penggunaan aplikasi “com.mobile.legends” terhadap VARK. Aplikasi ini merupakan aplikasi permainan strategi. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,005. Nilai tersebut lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi “com.mobile.legends” memiliki korelasi yang signifikan terhadap kelas A. Koefisien korelasi memiliki nilai -0.943 yang berarti kedua peubah ini memiliki korelasi yang sangat kuat dan negatif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi “com.mobile.legends” oleh seorang responden, maka semakin rendah kecenderungan preferensi responden tersebut terhadap informasi yang bersifat aural. Hasil berikutnya adalah sebagai berikut.

Tabel 4.16. Hasil Pengujian Durasi Penggunaan “com.pinterest”

Correlations ^a			duration	v	a	r	k
Spearman's rho	duration	Correlation Coefficient	1.000	.213	.841 [*]	-.377	-.101
		Sig. (2-tailed)	.	.686	.036	.461	.848
		N	6	6	6	6	6
	v	Correlation Coefficient	.213	1.000	.524	.277	.647
		Sig. (2-tailed)	.686	.	.286	.595	.165
		N	6	6	6	6	6
	a	Correlation Coefficient	.841 [*]	.524	1.000	.000	.429
		Sig. (2-tailed)	.036	.286	.	1.000	.396
		N	6	6	6	6	6
	r	Correlation Coefficient	-.377	.277	.000	1.000	.772
		Sig. (2-tailed)	.461	.595	1.000	.	.072
		N	6	6	6	6	6
	k	Correlation Coefficient	-.101	.647	.429	.772	1.000
		Sig. (2-tailed)	.848	.165	.396	.072	.
		N	6	6	6	6	6

*. Correlation is significant at the 0.05 level (2-tailed).

a. package_name = com.pinterest

Tabel 4.16 menampilkan hasil uji korelasi antara durasi penggunaan aplikasi “com.pinterest” terhadap VARK. Aplikasi ini merupakan aplikasi sosial media berbasis gambar dan video. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,036. Nilai tersebut lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi “com.pinterest” memiliki korelasi yang signifikan terhadap kelas A. Koefisien korelasi memiliki nilai 0.841 yang berarti kedua peubah ini memiliki korelasi yang sangat kuat dan positif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi “com.pinterest” oleh seorang responden, maka semakin tinggi kecenderungan preferensi responden tersebut terhadap informasi yang bersifat aural. Hasil berikutnya adalah sebagai berikut.

Tabel 4.17. Hasil Pengujian Durasi Penggunaan “com.samsung.android.video”

Correlations ^a			duration	v	a	r	k
Spearman's rho	duration	Correlation Coefficient	1.000	.048	-.193	-.810*	.115
		Sig. (2-tailed)	.	.909	.647	.015	.786
		N	8	8	8	8	8
	v	Correlation Coefficient	.048	1.000	.718*	-.085	.397
		Sig. (2-tailed)	.909	.	.045	.842	.331
		N	8	8	8	8	8
	a	Correlation Coefficient	-.193	.718*	1.000	.422	.523
		Sig. (2-tailed)	.647	.045	.	.298	.183
		N	8	8	8	8	8
	r	Correlation Coefficient	-.810*	-.085	.422	1.000	.153
		Sig. (2-tailed)	.015	.842	.298	.	.717
		N	8	8	8	8	8
	k	Correlation Coefficient	.115	.397	.523	.153	1.000
		Sig. (2-tailed)	.786	.331	.183	.717	.
		N	8	8	8	8	8

*. Correlation is significant at the 0.05 level (2-tailed).

a. package_name = com.samsung.android.video

Tabel 4.17. menampilkan hasil uji korelasi antara durasi penggunaan aplikasi “com.samsung.android.video” terhadap VARK. Aplikasi ini merupakan aplikasi pemutar video. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,015. Nilai tersebut lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi “com.samsung.android.video” memiliki korelasi yang signifikan terhadap kelas R. Koefisien korelasi memiliki nilai -0.81 yang berarti kedua peubah ini memiliki korelasi yang sangat kuat dan negatif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi “com.samsung.android.video” oleh seorang responden, maka semakin rendah kecenderungan preferensi responden tersebut terhadap informasi yang bersifat tekstual terstruktur. Hasil berikutnya adalah sebagai berikut.

Tabel 4.18. Hasil Pengujian Durasi Penggunaan “com.supercell.hayday”

Correlations ^a			duration	v	a	r	k
Spearman's rho	duration	Correlation Coefficient	1.000	-.100	.894*	.200	.600
		Sig. (2-tailed)	.	.873	.041	.747	.285
		N	5	5	5	5	5
	v	Correlation Coefficient	-.100	1.000	.112	.100	.700
		Sig. (2-tailed)	.873	.	.858	.873	.188
		N	5	5	5	5	5
	a	Correlation Coefficient	.894*	.112	1.000	.112	.783
		Sig. (2-tailed)	.041	.858	.	.858	.118
		N	5	5	5	5	5
	r	Correlation Coefficient	.200	.100	.112	1.000	.100
		Sig. (2-tailed)	.747	.873	.858	.	.873
		N	5	5	5	5	5
	k	Correlation Coefficient	.600	.700	.783	.100	1.000
		Sig. (2-tailed)	.285	.188	.118	.873	.
		N	5	5	5	5	5

*. Correlation is significant at the 0.05 level (2-tailed).

a. package_name = com.supercell.hayday

Tabel 4.x menampilkan hasil uji korelasi antara durasi penggunaan aplikasi “com.supercell.hayday” terhadap VARK. Aplikasi ini merupakan aplikasi permainan. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,041. Nilai tersebut lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi “com.supercell.hayday” memiliki korelasi yang signifikan terhadap kelas A. Koefisien korelasi memiliki nilai 0.894 yang berarti kedua peubah ini memiliki korelasi yang sangat kuat dan positif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi “com.supercell.hayday” oleh seorang responden, maka semakin tinggi kecenderungan preferensi responden tersebut terhadap informasi yang bersifat aural. Hasil berikutnya adalah sebagai berikut.

Tabel 4.19. Hasil Pengujian Frekuensi Penggunaan “cn.wps.moffice_eng”

Correlations ^a			frequency	v	a	r	k
Spearman's rho	frequency	Correlation Coefficient	1.000	.885**	.349	.627	.724*
		Sig. (2-tailed)	.	.003	.396	.096	.042
		N	8	8	8	8	8
	v	Correlation Coefficient	.885**	1.000	.301	.626	.513
		Sig. (2-tailed)	.003	.	.469	.097	.194
		N	8	8	8	8	8
	a	Correlation Coefficient	.349	.301	1.000	.256	.739*
		Sig. (2-tailed)	.396	.469	.	.540	.036
		N	8	8	8	8	8
	r	Correlation Coefficient	.627	.626	.256	1.000	.503
		Sig. (2-tailed)	.096	.097	.540	.	.204
		N	8	8	8	8	8
	k	Correlation Coefficient	.724*	.513	.739*	.503	1.000
		Sig. (2-tailed)	.042	.194	.036	.204	.
		N	8	8	8	8	8

** . Correlation is significant at the 0.01 level (2-tailed).

* . Correlation is significant at the 0.05 level (2-tailed).

a. package_name = cn.wps.moffice_eng

Tabel 4.19. menampilkan hasil uji korelasi antara frekuensi penggunaan aplikasi “cn.wps.moffice_eng” terhadap VARK. Aplikasi ini merupakan paket aplikasi Microsoft Office. Hasil menunjukkan bahwa durasi penggunaan dari aplikasi ini signifikan terhadap kelas V dan K. Adapun tingkat korelasi sangat kuat terhadap V dan tingkat kuat didapatkan terhadap K, keduanya dengan sifat positif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi “cn.wps.moffice_eng” oleh seorang responden, maka semakin tinggi kecenderungan preferensi responden tersebut terhadap informasi yang bersifat visual dan kinestetik. Hal ini cukup masuk akal jika melihat sifat aplikasi ini yang terdiri dari berbagai macam fungsi yang disediakan oleh Microsoft Office. Hasil berikutnya adalah sebagai berikut.

Tabel 4.20. Hasil Pengujian Frekuensi Penggunaan “com.android.camera”

Correlations ^a			frequency	v	a	r	k
Spearman's rho	frequency	Correlation Coefficient	1.000	-.608*	.097	-.210	-.498
		Sig. (2-tailed)	.	.047	.777	.535	.119
		N	11	11	11	11	11
	v	Correlation Coefficient	-.608*	1.000	.205	.671*	.577
		Sig. (2-tailed)	.047	.	.546	.024	.063
		N	11	11	11	11	11
	a	Correlation Coefficient	.097	.205	1.000	.238	.283
		Sig. (2-tailed)	.777	.546	.	.481	.400
		N	11	11	11	11	11
	r	Correlation Coefficient	-.210	.671*	.238	1.000	.419
		Sig. (2-tailed)	.535	.024	.481	.	.199
		N	11	11	11	11	11
	k	Correlation Coefficient	-.498	.577	.283	.419	1.000
		Sig. (2-tailed)	.119	.063	.400	.199	.
		N	11	11	11	11	11

*. Correlation is significant at the 0.05 level (2-tailed).

a. package_name = com.android.camera

Tabel 4.20 menampilkan hasil uji korelasi antara frekuensi penggunaan aplikasi “com.android.camera” terhadap VARK. Aplikasi ini merupakan aplikasi pengambil gambar dan video. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,047. Nilai tersebut lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi “com.android.camera” memiliki korelasi yang signifikan terhadap kelas V. Koefisien korelasi memiliki nilai -0.608 yang berarti kedua peubah ini memiliki korelasi yang kuat dan negatif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi “com.android.camera” oleh seorang responden, maka semakin rendah kecenderungan preferensi responden tersebut terhadap informasi yang bersifat visual. Hasil berikutnya adalah sebagai berikut.

Tabel 4.21. Hasil Pengujian Frekuensi Penggunaan “com.facebook.katana”

Correlations ^a			frequency	v	a	r	k
Spearman's rho	frequency	Correlation Coefficient	1.000	-.181	-.562	-.131	-.667 [*]
		Sig. (2-tailed)	.	.573	.057	.684	.018
		N	12	12	12	12	12
	v	Correlation Coefficient	-.181	1.000	-.014	.404	.276
		Sig. (2-tailed)	.573	.	.965	.192	.385
		N	12	12	12	12	12
	a	Correlation Coefficient	-.562	-.014	1.000	.099	.581 [*]
		Sig. (2-tailed)	.057	.965	.	.760	.048
		N	12	12	12	12	12
	r	Correlation Coefficient	-.131	.404	.099	1.000	.303
		Sig. (2-tailed)	.684	.192	.760	.	.339
		N	12	12	12	12	12
	k	Correlation Coefficient	-.667 [*]	.276	.581 [*]	.303	1.000
		Sig. (2-tailed)	.018	.385	.048	.339	.
		N	12	12	12	12	12

*. Correlation is significant at the 0.05 level (2-tailed).

a. package_name = com.facebook.katana

Tabel 4.21 menampilkan hasil uji korelasi antara frekuensi penggunaan aplikasi “com.facebook.katana” terhadap VARK. Aplikasi ini merupakan aplikasi sosial media Facebook. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,018. Nilai tersebut lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi “com.facebook.katana” memiliki korelasi yang signifikan terhadap kelas K. Koefisien korelasi memiliki nilai -0.667 yang berarti kedua peubah ini memiliki korelasi yang kuat dan negatif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi “com.facebook.katana” oleh seorang responden, maka semakin rendah kecenderungan preferensi responden tersebut terhadap informasi yang bersifat kinestetik. Hasil berikutnya adalah sebagai berikut.

Tabel 4.22. Hasil Uji Frekuensi Penggunaan “com.android.quicksearchbox”

Correlations ^a			frequency	v	a	r	k
Spearman's rho	frequency	Correlation Coefficient	1.000	-.273	-.771*	-.527	-.491
		Sig. (2-tailed)	.	.554	.043	.224	.263
		N	7	7	7	7	7
	v	Correlation Coefficient	-.273	1.000	.716	.618	.755*
		Sig. (2-tailed)	.554	.	.071	.139	.050
		N	7	7	7	7	7
	a	Correlation Coefficient	-.771*	.716	1.000	.495	.917**
		Sig. (2-tailed)	.043	.071	.	.258	.004
		N	7	7	7	7	7
	r	Correlation Coefficient	-.527	.618	.495	1.000	.309
		Sig. (2-tailed)	.224	.139	.258	.	.500
		N	7	7	7	7	7
	k	Correlation Coefficient	-.491	.755*	.917**	.309	1.000
		Sig. (2-tailed)	.263	.050	.004	.500	.
		N	7	7	7	7	7

*. Correlation is significant at the 0.05 level (2-tailed).

**. Correlation is significant at the 0.01 level (2-tailed).

a. package_name = com.android.quicksearchbox

Tabel 4.22 menampilkan hasil uji korelasi antara frekuensi penggunaan aplikasi “com.android.quicksearchbox” terhadap VARK. Aplikasi ini merupakan aplikasi penelusuran internet. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,043. Nilai tersebut lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi “com.android.quicksearchbox” memiliki korelasi yang signifikan terhadap kelas A. Koefisien korelasi memiliki nilai -0.771 yang berarti kedua peubah ini memiliki korelasi yang kuat dan negatif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi “com.android.quicksearchbox” oleh seorang responden, maka semakin rendah kecenderungan preferensi responden tersebut terhadap informasi yang bersifat aural.

Analisis uji korelasi pada skenario 1 ini mampu menggambarkan korelasi yang ada dari aplikasi-aplikasi tertentu secara spesifik terhadap kelas-kelas VARK. Pengamatan aplikasi secara bebas dan tanpa pengelompokan ini cukup baik untuk mengamati kelas aural pada VARK, karena 61% dari hasil pengujian menunjukkan adanya korelasi terhadap kelas aural. Terdapat pula kasus dimana

suatu aplikasi memiliki korelasi terhadap lebih dari 1 kelas. Kelas yang berkorelasi terhadap lebih dari 1 kelas VARK ini cukup menarik untuk diamati. Maka, dirancanglah uji korelasi skenario 2 untuk mengamati hal ini secara lebih rinci. Meski mampu menemukan sejumlah korelasi yang signifikan, namun masih terdapat sejumlah kelemahan dari skenario 1 ini. Terdapat sejumlah kasus dimana sejumlah aplikasi yang memiliki kesamaan fungsi diuji secara terpisah. Hal ini akan coba diatasi dan diamati pengaruhnya pada skenario 3 yang menggunakan proses pengelompokan aplikasi dalam pengamatannya.

4.2.3. Analisis Uji Korelasi Skenario 2

Pada skenario ini, aplikasi yang akan diuji dikelompokkan dalam kelas-kelas VARK yang sesuai dengan nilai VARK yang paling dominan dari responden. Sehingga, tiap aplikasi dapat diuji pada lebih dari 1 kelas VARK, tergantung pada responden dengan kelas dominan apa saja aplikasi tersebut muncul. Skenario ini mempersempit cakupan pengujian dengan menambahkan syarat baru. Pada skenario 1, kelompok data pengujian terdiri dari semua aplikasi dengan nama yang sama. Namun pada skenario ini, kelompok data pengujian hanya terdiri dari aplikasi dengan nama yang sama dan responden dengan nilai dominan VARK yang sama saja. Pengolahan data perlu dilakukan untuk mencari kelas VARK yang paling dominan dari tiap responden. Proses tersebut dilakukan melalui kode pemrograman yang ditampilkan pada Gambar 4.40.

```
#Daftar kelas VARK
categories = ['v', 'a', 'r', 'k']
#Daftar pola pengamatan
patterns = ['duration', 'frequency']

#Membaca data rekaman penggunaan aplikasi
df = pd.read_csv('1_statsReady.csv')
#Membaca data nilai aktual VARK responden
df2 = pd.read_csv('0_actualVARK.csv')

#Menemukan kelas paling dominan
df2['dominant_VARK'] = df2[categories].idxmax(axis=1)
#Menemukan nilai VARK tertinggi
df2['max_score'] = df2[categories].max(axis=1)
#Hapus kolom nilai VARK yang masih terpisah
df2.drop(columns=categories, inplace=True)
```

Gambar 4.40. Kode Pemrograman untuk Menentukan Kelas Paling Dominan

Pada proses ini, dibuat 2 kolom baru untuk menampung hasil pengolahan yang dilakukan. Kolom pertama menampung kelas VARK paling dominan dari seorang responden. Proses pencariannya dilakukan dengan fungsi “idxmax()” yang diterapkan pada nilai “axis”=1. Fungsi tersebut mencari indeks yang memiliki nilai tertinggi, kemudian parameter axis menentukan orientasi pencarian data. Nilai 1 memberitahu fungsi untuk melakukan pencarian secara mendatar. Kolom kedua menampung nilai tertinggi dari kelas VARK yang paling dominan. Fungsi yang digunakan untuk mencari nilai tersebut adalah “max()”. Setelah kedua informasi tersebut didapatkan, kolom nilai kelas-kelas VARK akan dihapus karena sudah tidak signifikan. Proses tersebut menghasilkan struktur data baru seperti yang ditampilkan pada Tabel 4.23.

Tabel 4.23. Tabel Nilai Dominan VARK

uid	dominant_VARK	max_score
2yzNg191GPW3nzLTGFviRc9VKKt2	k	11
39e9mvtpgFPx8grfQWhKB81tImv1	a	9
3ZkoMw38Tab7CvThKhYMYFh8rcu1	v	9
5ADMMnGeWxXnLjx2PuyFQ1Y0Deu2	r	10
7v3Xow5LioPI9rG1ym6vKI8N9kJ2	r	9
D4NvI0aWY7ZvRaxIkrWwk9XDftA3	k	13
DUwqt99Br9QkLepIeDNyA75snZe2	a	6
FL1xLv67JdeITPXCy0gbvJYjpUf2	r	7
FbIUf526beR7ihXnUXcLqAjVUql2	k	12
Fy8YbL5UsAfv98QO3xkctweKRbV2	a	13
HbKn5tMzX2czz3BfQlKMTK4npe93	k	9
JqGgrcVjMDW0Qxd9qxpeuzT9pCm1	k	11
Kn959m2CGZujZMeidu6RITUdFI2	v	8

Setelah tabel nilai dominan VARK terbentuk, maka perlu digabungkan dengan tabel rekaman penggunaan aplikasi untuk membuat *dataset* pengujian. Proses pembentukan *dataset* pengujian untuk skenario 2 ditampilkan pada Gambar 4.41.

```

#Menuliskan hasil ke dalam format .xlsx
writer = pd.ExcelWriter('Dataset.xlsx')

#Menghapus aplikasi yang tidak signifikan
df = df[~(df.package_name.isin(excluded))].reset_index(drop=True)

#Fungsi pembentukan dataset uji korelasi
def build_testDf(pola, kategori):
    #Menghapus nilai 0 pada kolom pola yang dipilih
    df_ = df.loc[(df[pola]!=0), ['uid','package_name',pola]]
    #Menggabungkan data ukuran pola dengan nilai VARK aktual
    merged = pd.merge(df_, df2, on='uid')
    #Menyaring aplikasi berdasarkan kelas dominan yang akan diamati
    filtered = merged[merged.dominant_VARK==kategori]
    #Menghapus aplikasi-aplikasi dengan jumlah sampel yang terlalu sedikit
    dataset = filtered.groupby('package_name').filter(lambda x: len(x) > 4)
    #Menerapkan fungsi normalisasi pada kolom pola untuk tiap pengguna (uid)
    dataset[pola] = dataset.groupby('uid')[pola].apply(lambda x: (x-min(x))/(max(x)-min(x)))
    #Menuliskan data ke dalam sheet excel
    dataset.to_excel(writer,pola+'_'+kategori, index=False)

```

Gambar 4.41. Pemrograman Pembentukan *Dataset* Skenario 2

Secara garis besar, fungsi pembentukan *dataset* cukup mirip dengan fungsi pembentukan *dataset* pada skenario 1. Namun, kali ini terdapat 2 parameter, yaitu pola dan kategori pengamatan. Dibutuhkan 2 parameter karena kali ini pengujian akan dilakukan berdasarkan kategori aplikasi. Sehingga, pada tiap iterasi akan dibentuk *dataset* pengamatan dari kombinasi untuk tiap pola dan kategori pengamatan yang terpilah-pilah dalam bentuk *sheet*. Normalisasi terhadap nilai pola pengamatan juga dilakukan untuk tiap pengguna dalam satu *sheet* pengamatan. Tabel 4.24. menampilkan struktur data pada *sheet* pengamatan durasi dari aplikasi dengan kategori visual.

Tabel 4.24. *Dataset* Pengujian Skenario 2

uid	package_name	duration	dominant_VARK	max_score
3ZkoMw38Tab7CvThKhYMYFh8rcu1	com.android.chrome	0.021030627 v		9
3ZkoMw38Tab7CvThKhYMYFh8rcu1	com.google.android.apps.docs	0 v		9
3ZkoMw38Tab7CvThKhYMYFh8rcu1	com.google.android.apps.maps	0.042889197 v		9
3ZkoMw38Tab7CvThKhYMYFh8rcu1	com.google.android.gm	0.012622406 v		9
3ZkoMw38Tab7CvThKhYMYFh8rcu1	com.google.android.youtube	0.42476189 v		9
3ZkoMw38Tab7CvThKhYMYFh8rcu1	com.instagram.android	1 v		9
3ZkoMw38Tab7CvThKhYMYFh8rcu1	com.path	0.515040087 v		9
3ZkoMw38Tab7CvThKhYMYFh8rcu1	com.tokopedia.tkpd	0.360566085 v		9
3ZkoMw38Tab7CvThKhYMYFh8rcu1	com.traveloka.android	0.760815607 v		9
3ZkoMw38Tab7CvThKhYMYFh8rcu1	com.whatsapp	0.305349738 v		9

Dataset yang terbentuk tersebut kemudian dimasukkan ke dalam SPSS untuk analisis uji korelasi. Analisis uji korelasi dengan skenario ini juga mampu menemukan sejumlah korelasi antara data penggunaan aplikasi *smartphone*

dengan kelas-kelas VARK. Adapun hasil yang berhasil ditemukan akan dijabarkan sebagai berikut.

Tabel 4.25. Uji Durasi “com.google.android.youtube” terhadap Kelas A

Correlations ^a			duration	max_score
Spearman's rho	duration	Correlation Coefficient	1.000	.955**
		Sig. (2-tailed)	.	.001
		N	7	7
	max_score	Correlation Coefficient	.955**	1.000
		Sig. (2-tailed)	.001	.
		N	7	7

** . Correlation is significant at the 0.01 level (2-tailed).

a. package_name = com.google.android.youtube

Tabel 4.25 menampilkan hasil uji korelasi antara durasi penggunaan aplikasi “com.google.android.youtube” oleh pengguna dengan kelas dominan A, terhadap kelas A dari nilai aktual VARK. Aplikasi ini merupakan aplikasi berbagi video daring. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,001. Nilai tersebut lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi “com.google.android.youtube” memiliki korelasi yang signifikan terhadap kelas A. Koefisien korelasi memiliki nilai 0.955 yang berarti kedua peubah ini memiliki korelasi yang sangat kuat dan bersifat positif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi “com.google.android.youtube” oleh seorang responden dengan kelas dominan A, maka semakin tinggi kecenderungan preferensi responden tersebut terhadap informasi yang bersifat aural. Hasil selanjutnya yang didapatkan adalah sebagai berikut.

Tabel 4.26. Uji Durasi “com.microsoft.office.powerpoint” terhadap Kelas K

Correlations ^a			duration	max_score
Spearman's rho	duration	Correlation Coefficient	1.000	-.883 [*]
		Sig. (2-tailed)	.	.020
		N	6	6
	max_score	Correlation Coefficient	-.883 [*]	1.000
		Sig. (2-tailed)	.020	.
		N	6	6

*. Correlation is significant at the 0.05 level (2-tailed).

a. package_name = com.microsoft.office.powerpoint

Tabel 4.26 menampilkan hasil uji korelasi antara durasi penggunaan aplikasi “com.microsoft.office.powerpoint” oleh pengguna dengan kelas dominan K, terhadap kelas K dari nilai aktual VARK. Aplikasi ini merupakan aplikasi pembaca dan penyusun presentasi. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,001. Nilai tersebut lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi “com.microsoft.office.powerpoint” memiliki korelasi yang signifikan terhadap kelas K. Koefisien korelasi memiliki nilai -0.883 yang berarti kedua peubah ini memiliki korelasi yang sangat kuat dan bersifat negatif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi “com.microsoft.office.powerpoint” oleh seorang responden dengan kelas dominan K, maka semakin tinggi kecenderungan preferensi responden tersebut terhadap informasi yang bersifat kinestetik. Hasil selanjutnya yang didapatkan adalah sebagai berikut.

Tabel 4.27. Uji Frekuensi “com.android.phone” terhadap Kelas A

Correlations^a

			frequency	max_score
Spearman's rho	frequency	Correlation Coefficient	1.000	.805 [*]
		Sig. (2-tailed)	.	.016
		N	8	8
	max_score	Correlation Coefficient	.805 [*]	1.000
		Sig. (2-tailed)	.016	.
		N	8	8

*. Correlation is significant at the 0.05 level (2-tailed).

a. package_name = com.android.phone

Tabel 4.27 menampilkan hasil uji korelasi antara frekuensi penggunaan aplikasi “com.android.phone” oleh pengguna dengan kelas dominan A, terhadap kelas A dari nilai aktual VARK. Aplikasi ini merupakan aplikasi pembaca dan penyusun presentasi. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,02. Nilai tersebut lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi “com.android.phone” memiliki korelasi yang signifikan terhadap kelas A. Koefisien korelasi memiliki nilai 0.805 yang berarti kedua peubah ini memiliki korelasi yang sangat kuat dan bersifat positif. Hal ini menunjukkan bahwa semakin tinggi frekuensi penggunaan aplikasi “com.android.phone” oleh seorang responden dengan kelas dominan A, maka semakin tinggi kecenderungan preferensi responden tersebut terhadap informasi yang bersifat aural. Hasil selanjutnya yang didapatkan adalah sebagai berikut.

Tabel 4.28. Uji Frekuensi “com.google.android.apps.maps” terhadap Kelas K

Correlations^a

			frequency	max_score
Spearman's rho	frequency	Correlation Coefficient	1.000	.661 [*]
		Sig. (2-tailed)	.	.027
		N	11	11
	max_score	Correlation Coefficient	.661 [*]	1.000
		Sig. (2-tailed)	.027	.
		N	11	11

*. Correlation is significant at the 0.05 level (2-tailed).

a. package_name = com.google.android.apps.maps

Tabel 4.28 menampilkan hasil uji korelasi antara frekuensi penggunaan aplikasi “com.google.android.apps.maps” oleh pengguna dengan kelas dominan K, terhadap kelas K dari nilai aktual VARK. Aplikasi ini merupakan aplikasi penampil peta dan layanan navigasi. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,027. Nilai tersebut lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi “com.google.android.apps.maps” memiliki korelasi yang signifikan terhadap kelas K. Koefisien korelasi memiliki nilai 0.661 yang berarti kedua peubah ini memiliki korelasi yang kuat dan bersifat positif. Hal ini menunjukkan bahwa semakin tinggi frekuensi penggunaan aplikasi “com.google.android.apps.maps” oleh seorang responden dengan kelas dominan K, maka semakin tinggi kecenderungan preferensi responden tersebut terhadap informasi yang bersifat kinestetik.

Korelasi yang didapatkan dari skenario ini sangat sedikit jika dibandingkan dengan skenario sebelumnya, namun tingkat korelasi yang dihasilkan sangat memuaskan. Sebesar 75% dari korelasi signifikan yang ditemukan, memiliki tingkat korelasi yang sangat kuat. Selain itu, skenario ini juga baik untuk digunakan dalam pengamatan korelasi yang bersifat positif, karena 75% dari korelasi signifikan yang ditemukan melalui skenario ini memiliki sifat positif.

Akan tetapi, penerapan skenario ini sangatlah terbatas karena dibatasi oleh kelas VARK dominan dari responden.

4.2.4. Analisis Uji Korelasi Skenario 3

Pada skenario ini, aplikasi yang akan diuji dikelompokkan berdasarkan fungsinya. Skenario ini dirancang untuk mengatasi kelemahan skenario 1 dengan melakukan pengelompokan, namun tetap mempertahankan luasnya cakupan dan kemungkinan hasil penelitian. Pengelompokan aplikasi dilakukan melalui 2 tahap. Tahap pertama adalah pengelompokan otomatis berdasarkan kategori aplikasi yang tertera pada Google Play Store. Kode pemrograman untuk pengelompokan otomatis ditampilkan pada Gambar 4.42.

```
#2. Penggalan dan penambahan data kategori aplikasi

import play_scraper as ps #pustaka penggalan data aplikasi dari Google Play Store

#Mendefinisikan variabel penampung data kategori aplikasi
categories = []

#Loop penggalan data dari semua aplikasi yang terdapat pada data
for name in usageStats.package_name:
    try:
        #Menggali semua informasi yang tersedia dari suatu aplikasi
        details = ps.details(name)
        #Menggali data kategori aplikasi saja; menggunakan index karena berbentuk list
        category = (details.get('category'))[0]
    except:
        #Jika aplikasi tidak ditemukan, kategori=null
        category = None
    #Memasukkan data kategori ke dalam daftar
    categories.append(category)

#Membangun kolom baru yang berisikan daftar kategori aplikasi
usageStats['app_category'] = categories
#Menuliskan data ke dalam format .csv; masih membutuhkan pengolahan kategori lebih lanjut
usageStats.to_csv('1_statsClear_categoryFixed.csv', index=False)
```

Gambar 4.42. Kode Pemrograman untuk Pengelompokan Aplikasi

Kategori aplikasi didapatkan dengan melakukan penggalan data dari laman situs Google Play Store. Pustaka yang digunakan untuk melakukan penggalan data adalah play_scrapper. Pustaka ini mampu menggali data berupa daftar aplikasi yang tersedia, rincian aplikasi, daftar kategori yang tersedia, dll. Data yang akan digali pada penelitian ini adalah data rincian aplikasi. Rincian aplikasi berisi berbagai macam informasi seperti harga, *rating*, pengembang, deskripsi, kategori, dll. Dari tiap rincian, hanya kategori aplikasi yang akan diambil. Data kategori

aplikasi yang didapatkan akan disimpan ke dalam peubah “categories” yang bertipe *list*. Ketika suatu aplikasi tidak dapat ditemukan pada Google Play Store, maka kategorinya dianggap *null*. Penggalan data dilakukan secara iteratif untuk tiap aplikasi yang terdapat pada kolom “package_name”. Setelah iterasi penggalan data selesai dilakukan, peubah “categories” yang menyimpan daftar kategori aplikasi akan dimasukkan sebagai kolom baru pada tabel rekaman penggunaan aplikasi. Hasil dari proses ini ditampilkan pada Tabel 4.29.

Tabel 4.29. Tabel Hasil Pengolahan Data Tahap Kedua

uid	package_name	duration	frequency	app_category
2yzNg191GPW3nzLTGFvIRc9VKKt2	com.madcat.easyposer	28501	0	ART_AND_DESIGN
xSQQR72qePFnGFFcS4imXvh2na2	com.canva.editor	512803	16	ART_AND_DESIGN
UUKclXyfiFYGaTk3dHCRAWP62gO2	jp.co.a_tm.android.plus_nakayoshi_kaeru	5890	0	ART_AND_DESIGN
bEOeSmcGWAMjh39RxtwvZTvXtEc2	com.canva.editor	6336250	0	ART_AND_DESIGN
fUcLIcEuCtMSd93yERQrJLqOMp53	com.sonymobile.sketch	127157	13	ART_AND_DESIGN
FL1xLv67JdeITPXcy0gbvYjpUf2	com.samsung.android.app.bikemode	65522	0	AUTO_AND_VEHICLES
vd8wf5ehIP6SM3GnWTHOHTPJJaS2	com.samsung.android.app.bikemode	10815	0	AUTO_AND_VEHICLES
HbKn5tMzX2czz3BfQlKMTK4npe93	com.samsung.android.app.bikemode	165683	4	AUTO_AND_VEHICLES
oi4NtYZAJ7PvbyxHN3EfQX5J4QT2	com.C1TutorialMakeupEyes	91747	0	BEAUTY

Tahap kedua adalah melakukan pengelompokan aplikasi secara manual dengan mengacu pada hasil pengelompokan otomatis. Proses ini pertama-tama dilakukan dengan melakukan pengelompokan terhadap aplikasi-aplikasi yang belum memiliki kategori karena tidak tersedia pada Google Play Store. Kemudian dilakukan pula sejumlah pemecahan dan penggabungan kategori. Sebagai contoh, kategori “Video Players & Editors” dapat dipecah menjadi “Video Player” dan “Video Editor”. Kategori “News & Magazines”, “Books & Reference”, dan aplikasi-aplikasi pencatat memo dapat digabungkan menjadi “Read/Write”. Selain itu, dilakukan pula penghapusan terhadap kategori-kategori yang tidak signifikan. Kategori yang dilibatkan dalam penelitian ini hanyalah kategori-kategori yang sekiranya menyediakan informasi bagi pengguna atau memicu pengguna untuk mempelajari sesuatu. Sehingga, aplikasi-aplikasi yang tidak memiliki fungsionalitas tersebut akan dihapus, contohnya aplikasi dengan kategori “Tools”. Proses ini menghasilkan daftar kategori baru dengan rincian yang tertera pada Tabel 4.30. Data yang telah dikelompokkan melalui proses ini akan diolah lebih lanjut pada tahap terakhir.

Tabel 4.30. Tabel Rincian Kategori Aplikasi

Kategori	Rincian
AUDIO_EDITOR_RECORDER	Perekam, penyunting, dan konfigurasi audio.
AUDIO_PLAYER	Pemutar audio.
BROWSER	Peramban.
CALCULATOR	Kalkulator dan alat konversi satuan pengukuran.
CALENDAR	Kalender.
COMICS	Pembaca konten yang berupa kombinasi dari gambar dan teks, seperti buku komik, <i>manga</i> , dan <i>meme</i> .
COMMUNICATION	Alat komunikasi seperti telepon, <i>e-mail</i> , sosial media, forum, dan pengirim pesan.
DICTIONARY	Kamus.
EDUCATION	Media pembelajaran seperti <i>e-learning</i> , permainan edukasi, dll.
FINANCE	Pencatat dan pengolahan keuangan.
GAME_PUZZLE_BOARD	Permainan berbasis pola visual.
GAME_CASUAL	Permainan umum.
GAME_MUSIC	Permainan yang berkaitan dengan musik.
GAME_SIMULATION	Permainan simulasi dan <i>role play</i> .
GAME_SPORTS	Permainan yang berkaitan dengan olahraga.
GAME_STRATEGY	Permainan yang erat dengan penyusunan strategi.
GAME_WORD_QUIZ	Permainan yang berkaitan erat dengan pengolahan kata.
HEALTH_AND_FITNESS	Pemantau dan pemandu kegiatan olahraga dan kesehatan.
MAPS_AND_NAVIGATION	Penampil peta dan layanan navigasi.
READ_WRITE	Pembaca dan pengolah konten tekstual.
VISUAL_CAPTURER	Perekam konten visual seperti foto, video, dan <i>scanner</i> .
VISUAL_EDITOR	Penyunting, perancang, dan pengolahan konten visual.
VISUAL_VIEWER	Penampil konten visual.
WEATHER	Penampil informasi cuaca.

Tahap terakhir pada pengolahan awal adalah normalisasi data dan pembentukan data pengujian. Adapun kode pemrograman untuk tahap ini ditampilkan pada Gambar 4.43.

```
#3. Pembentukan dataset uji korelasi

#Hapus semua baris yang memiliki missing value
df.dropna(inplace=True)

#Menuliskan hasil ke dalam format .xlsx
writer = pd.ExcelWriter('Dataset3.xlsx')

#Fungsi pembentukan dataset uji korelasi
def build_testDf(pola):
    #Menghapus nilai 0 pada kolom pola yang dipilih
    filtered = df.loc[(df[pola]!=0), ['uid', 'package_name', 'app_category', pola]]
    #Menerapkan fungsi normalisasi pada kolom pola untuk tiap pengguna (uid)
    filtered[pola] = filtered.groupby('uid')[pola].apply(lambda x: (x-min(x))/(max(x)-min(x)))
    #Menentukan jumlah total nilai pola dari tiap kategori aplikasi yang dimiliki tiap pengguna
    total = filtered.groupby(['uid', 'app_category'], as_index=False)[pola].sum()
    #Membatasi jumlah sampel
    total = total.groupby(['app_category']).filter(lambda x: len(x) > 4)
    #Menggabungkan data ukuran pola dengan nilai VARK aktual
    dataset = pd.merge(total, df2, on='uid')
    #Menuliskan data ke dalam sheet excel
    dataset.to_excel(writer, pola, index=False)
```

Gambar 4.43. Kode Pemrograman untuk Pembentukan Data Pengujian

Pengolahan data yang akan dilakukan pada tahap ini didefinisikan pada suatu fungsi dengan parameter berupa pola dan kategori aplikasi yang akan diamati. Pertama-tama kembali dilakukan kembali pembersihan data dari nilai *null* pada kategori aplikasi dan nilai 0 pada pola yang akan diamati. Kemudian diterapkan fungsi normalisasi pada nilai-nilai pola pengamatan yang dikelompokkan berdasarkan responden yang bersangkutan. Nilai pola yang telah ternormalisasi kemudian dijumlahkan per kategori untuk tiap responden. Tabel rekaman penggunaan aplikasi yang terbentuk kemudian dipilah-pilah untuk tiap kategori, dan digabungkan dengan data nilai VARK aktual. Penggabungan data dilakukan dengan fungsi *merge*. Fungsi tersebut secara *default* menggabungkan data secara mendatar. Hasil pemilahan akan disimpan sebagai sejumlah *sheet* dalam suatu *file* Excel. Gambar 4.31 memperlihatkan struktur data dari data pengujian durasi “AUDIO_PLAYER”. Data pengujian yang telah terbentuk akan digunakan dalam analisis uji korelasi.

Tabel 4.31. Pengujian Durasi “AUDIO_PLAYER”

uid	app_category	duration	v	a	r	k
2yzNg191t	AUDIO_PLAYER	0.003881	9	7	10	11
39e9mvtp	AUDIO_PLAYER	0.000357	8	9	8	4
3ZkoMw3i	AUDIO_PLAYER	0.049402	9	1	8	4
7v3Xow5L	AUDIO_PLAYER	0.004882	8	2	9	5
FL1xLv67Ji	AUDIO_PLAYER	0.024649	2	1	7	7
FbiUf526b	AUDIO_PLAYER	0.004245	8	7	9	12

Dari semua kombinasi pola dan kategori aplikasi yang diuji, ditemukan sejumlah korelasi yang signifikan pada:

- a. Durasi “CALCULATOR”
- b. Durasi “DICTIONARY”
- c. Durasi “EDUCATION”
- d. Durasi “GAME_SIMULATION”
- e. Durasi “VISUAL_EDITOR”
- f. Frekuensi “COMMUNICATION”
- g. Frekuensi “GAME_STRATEGY”
- h. Frekuensi “WEATHER”

Adapun penjabaran dari setiap korelasi signifikan yang ditemukan akan dijabarkan sebagai berikut.

Tabel 4.32. Hasil Uji Korelasi pada Durasi “CALCULATOR”

Correlations			duration	v	a	r	k
Spearman's rho	duration	Correlation Coefficient	1.000	-.370*	-.066	-.202	-.074
		Sig. (2-tailed)	.	.029	.708	.245	.671
		N	35	35	35	35	35
	v	Correlation Coefficient	-.370*	1.000	.331	.367*	.381*
		Sig. (2-tailed)	.029	.	.052	.030	.024
		N	35	35	35	35	35
	a	Correlation Coefficient	-.066	.331	1.000	.284	.535**
		Sig. (2-tailed)	.708	.052	.	.099	.001
		N	35	35	35	35	35
	r	Correlation Coefficient	-.202	.367*	.284	1.000	.181
		Sig. (2-tailed)	.245	.030	.099	.	.299
		N	35	35	35	35	35
	k	Correlation Coefficient	-.074	.381*	.535**	.181	1.000
		Sig. (2-tailed)	.671	.024	.001	.299	.
		N	35	35	35	35	35

*. Correlation is significant at the 0.05 level (2-tailed).

**. Correlation is significant at the 0.01 level (2-tailed).

Tabel 4.32 menampilkan hasil uji korelasi antara durasi penggunaan aplikasi “CALCULATOR” terhadap VARK. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,029. Nilai ini lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi dengan kategori “CALCULATOR” memiliki korelasi yang signifikan terhadap kelas V. Koefisien korelasi memiliki nilai -0,37 yang berarti kedua peubah ini memiliki korelasi yang bersifat lemah dan negatif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi komputasi oleh seorang responden, maka semakin rendah preferensi responden tersebut terhadap konten-konten pembelajaran visual.

Tabel 4.33. Tabel Hasil Uji Korelasi pada Durasi “DICTIONARY”

Correlations			duration	v	a	r	k
Spearman's rho	duration	Correlation Coefficient	1.000	-.406	-.396	-.685**	-.309
		Sig. (2-tailed)	.	.119	.129	.003	.245
		N	16	16	16	16	16
	v	Correlation Coefficient	-.406	1.000	.172	.343	.469
		Sig. (2-tailed)	.119	.	.523	.194	.067
		N	16	16	16	16	16
	a	Correlation Coefficient	-.396	.172	1.000	.394	.578*
		Sig. (2-tailed)	.129	.523	.	.131	.019
		N	16	16	16	16	16
	r	Correlation Coefficient	-.685**	.343	.394	1.000	.318
		Sig. (2-tailed)	.003	.194	.131	.	.230
		N	16	16	16	16	16
	k	Correlation Coefficient	-.309	.469	.578*	.318	1.000
		Sig. (2-tailed)	.245	.067	.019	.230	.
		N	16	16	16	16	16

** . Correlation is significant at the 0.01 level (2-tailed).

* . Correlation is significant at the 0.05 level (2-tailed).

Tabel 4.33 menampilkan hasil uji korelasi antara durasi penggunaan aplikasi “DICTIONARY” terhadap VARK. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,003. Nilai ini lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi dengan kategori “DICTIONARY” memiliki korelasi yang signifikan terhadap kelas R. Koefisien korelasi memiliki nilai -0,685 yang berarti kedua peubah ini memiliki korelasi yang bersifat kuat dan negatif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi kamus oleh seorang responden, maka semakin rendah preferensi responden tersebut terhadap konten-konten pembelajaran tekstual terstruktur. Tentu dibutuhkan pengamatan lebih dalam untuk menjelaskan faktor-faktor yang mempengaruhi sifat hubungan kedua peubah ini. Namun, dapat diasumsikan bahwa ketika seseorang menggunakan kamus terlalu lama, terdapat kemungkinan bahwa orang tersebut tidak cocok atau tidak terbiasa dengan konten tekstual yang terdapat pada kamus. Sehingga, orang tersebut membutuhkan waktu lebih banyak untuk memahami arti atau mencari suatu kata.

Tabel 4.34. Tabel Hasil Uji Korelasi pada Durasi “EDUCATION”

Correlations			duration	v	a	r	k
Spearman's rho	duration	Correlation Coefficient	1.000	-.050	-.609*	-.053	-.353
		Sig. (2-tailed)	.	.878	.036	.869	.261
		N	12	12	12	12	12
	v	Correlation Coefficient	-.050	1.000	-.022	.763**	.384
		Sig. (2-tailed)	.878	.	.947	.004	.218
		N	12	12	12	12	12
	a	Correlation Coefficient	-.609*	-.022	1.000	.142	.482
		Sig. (2-tailed)	.036	.947	.	.660	.112
		N	12	12	12	12	12
	r	Correlation Coefficient	-.053	.763**	.142	1.000	.606*
		Sig. (2-tailed)	.869	.004	.660	.	.037
		N	12	12	12	12	12
	k	Correlation Coefficient	-.353	.384	.482	.606*	1.000
		Sig. (2-tailed)	.261	.218	.112	.037	.
		N	12	12	12	12	12

*. Correlation is significant at the 0.05 level (2-tailed).

**. Correlation is significant at the 0.01 level (2-tailed).

Tabel 4.34 menampilkan hasil uji korelasi antara durasi penggunaan aplikasi “EDUCATION” terhadap VARK. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,036. Nilai ini lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi dengan kategori “EUCATION” memiliki korelasi yang signifikan terhadap kelas A. Koefisien korelasi memiliki nilai -0,609 yang berarti kedua peubah ini memiliki korelasi yang bersifat kuat dan negatif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi yang berkaitan dengan edukasi, maka semakin rendah preferensi responden terkait terhadap konten-konten pembelajaran yang bersifat aural.

Tabel 4.35. Tabel Hasil Uji Korelasi pada Durasi “GAME_SIMULATION”

Correlations			duration	v	a	r	k
Spearman's rho	duration	Correlation Coefficient	1.000	.260	.401	.022	.466*
		Sig. (2-tailed)	.	.269	.080	.927	.038
		N	20	20	20	20	20
	v	Correlation Coefficient	.260	1.000	.310	.211	.317
		Sig. (2-tailed)	.269	.	.184	.372	.173
		N	20	20	20	20	20
	a	Correlation Coefficient	.401	.310	1.000	.078	.540*
		Sig. (2-tailed)	.080	.184	.	.742	.014
		N	20	20	20	20	20
	r	Correlation Coefficient	.022	.211	.078	1.000	.131
		Sig. (2-tailed)	.927	.372	.742	.	.581
		N	20	20	20	20	20
	k	Correlation Coefficient	.466*	.317	.540*	.131	1.000
		Sig. (2-tailed)	.038	.173	.014	.581	.
		N	20	20	20	20	20

*. Correlation is significant at the 0.05 level (2-tailed).

Tabel 4.35 menampilkan hasil uji korelasi antara durasi penggunaan aplikasi “GAME_SIMULATION” terhadap VARK. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,038. Nilai ini lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi dengan kategori “GAME_SIMULATION” memiliki korelasi yang signifikan terhadap kelas K. Koefisien korelasi memiliki nilai 0,466 yang berarti kedua peubah ini memiliki korelasi yang bersifat lumayan kuat dan positif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi permainan simulasi oleh seorang responden, maka semakin tinggi preferensi responden tersebut terhadap konten-konten pembelajaran yang bersifat kinestetik. Tentu dibutuhkan pengamatan lebih dalam untuk menjelaskan faktor-faktor yang mempengaruhi sifat hubungan kedua peubah ini. Namun, secara konsep memang terdapat kesesuaian sifat pada kedua peubah ini. Kinestetik dalam VARK mencakup konten-konten atau kegiatan-kegiatan yang bersifat praktis, salah satunya adalah dengan melakukan simulasi. Hal yang sama juga terjadi pada penggunaan aplikasi permainan simulasi, dimana pengguna dihadapkan pada situasi yang menyerupai kondisi aslinya di dunia nyata.

Tabel 4.36. Tabel Hasil Uji Korelasi pada Durasi “VISUAL_EDITOR”

Correlations			duration	v	a	r	k
Spearman's rho	duration	Correlation Coefficient	1.000	-.185	-.140	-.523**	-.088
		Sig. (2-tailed)	.	.301	.436	.002	.625
		N	33	33	33	33	33
	v	Correlation Coefficient	-.185	1.000	.365*	.356*	.410*
		Sig. (2-tailed)	.301	.	.037	.042	.018
		N	33	33	33	33	33
	a	Correlation Coefficient	-.140	.365*	1.000	.323	.594**
		Sig. (2-tailed)	.436	.037	.	.067	.000
		N	33	33	33	33	33
	r	Correlation Coefficient	-.523**	.356*	.323	1.000	.208
		Sig. (2-tailed)	.002	.042	.067	.	.245
		N	33	33	33	33	33
	k	Correlation Coefficient	-.088	.410*	.594**	.208	1.000
		Sig. (2-tailed)	.625	.018	.000	.245	.
		N	33	33	33	33	33

** . Correlation is significant at the 0.01 level (2-tailed).

* . Correlation is significant at the 0.05 level (2-tailed).

Tabel 4.36 menampilkan hasil uji korelasi antara durasi penggunaan aplikasi “VISUAL_EDITOR” terhadap VARK. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,002. Nilai ini lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa durasi penggunaan aplikasi dengan kategori “VISUAL_EDITOR” memiliki korelasi yang signifikan terhadap kelas R. Koefisien korelasi memiliki nilai -0,523 yang berarti kedua peubah ini memiliki korelasi yang bersifat lumayan kuat dan negatif. Hal ini menunjukkan bahwa semakin tinggi durasi penggunaan aplikasi penyunting konten visual oleh seorang responden, maka semakin rendah preferensi responden tersebut terhadap konten-konten pembelajaran tekstual terstruktur.

Tabel 4.37. Tabel Hasil Uji Korelasi pada Frekuensi “COMMUNICATION”

Correlations			frequency	v	a	r	k
Spearman's rho	frequency	Correlation Coefficient	1.000	-.138	-.268	-.013	-.440**
		Sig. (2-tailed)	.	.421	.114	.942	.007
		N	36	36	36	36	36
	v	Correlation Coefficient	-.138	1.000	.346*	.376*	.397*
		Sig. (2-tailed)	.421	.	.039	.024	.017
		N	36	36	36	36	36
	a	Correlation Coefficient	-.268	.346*	1.000	.300	.548**
		Sig. (2-tailed)	.114	.039	.	.075	.001
		N	36	36	36	36	36
	r	Correlation Coefficient	-.013	.376*	.300	1.000	.197
		Sig. (2-tailed)	.942	.024	.075	.	.250
		N	36	36	36	36	36
	k	Correlation Coefficient	-.440**	.397*	.548**	.197	1.000
		Sig. (2-tailed)	.007	.017	.001	.250	.
		N	36	36	36	36	36

** . Correlation is significant at the 0.01 level (2-tailed).

* . Correlation is significant at the 0.05 level (2-tailed).

Tabel 4.37 menampilkan hasil uji korelasi antara frekuensi penggunaan aplikasi “COMMUNACTION” terhadap VARK. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,007. Nilai ini lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa frekuensi penggunaan aplikasi dengan kategori “COMMUNICATION” memiliki korelasi yang signifikan terhadap kelas K. Koefisien korelasi memiliki nilai -0.44 yang berarti kedua peubah ini memiliki korelasi yang bersifat lumayan kuat dan negatif. Hal ini menunjukkan bahwa semakin tinggi frekuensi penggunaan aplikasi komunikasi oleh seorang responden, maka semakin rendah preferensi responden tersebut terhadap konten-konten pembelajaran yang bersifat kinestetik.

Tabel 4.38. Hasil Uji Korelasi pada Frekuensi “WEATHER”

Correlations ^a			frequency	v	a	r	k
Spearman's rho	frequency	Correlation Coefficient	1.000	-.348	.348	-.257	-.841*
		Sig. (2-tailed)	.	.499	.499	.623	.036
		N	6	6	6	6	6
	v	Correlation Coefficient	-.348	1.000	-.176	.754	.279
		Sig. (2-tailed)	.499	.	.738	.084	.592
		N	6	6	6	6	6
	a	Correlation Coefficient	.348	-.176	1.000	-.029	.132
		Sig. (2-tailed)	.499	.738	.	.957	.803
		N	6	6	6	6	6
	r	Correlation Coefficient	-.257	.754	-.029	1.000	.203
		Sig. (2-tailed)	.623	.084	.957	.	.700
		N	6	6	6	6	6
	k	Correlation Coefficient	-.841*	.279	.132	.203	1.000
		Sig. (2-tailed)	.036	.592	.803	.700	.
		N	6	6	6	6	6

*. Correlation is significant at the 0.05 level (2-tailed).

a. app_category = WEATHER

Tabel 4.38 menampilkan hasil uji korelasi antara frekuensi penggunaan aplikasi “WEATHER” terhadap VARK. Nampak bahwa tingkat signifikansi dari korelasi kedua peubah tersebut adalah 0,036. Nilai ini lebih kecil dari batas tingkat signifikansi yang ditetapkan, yaitu 0,05. Hal ini menunjukkan bahwa frekuensi penggunaan aplikasi dengan kategori “WEATHER” memiliki korelasi yang signifikan terhadap kelas K. Koefisien korelasi memiliki nilai -0.841 yang berarti kedua peubah ini memiliki korelasi yang bersifat sangat kuat dan negatif. Hal ini menunjukkan bahwa semakin tinggi frekuensi penggunaan aplikasi penampil cuaca oleh seorang responden, maka semakin rendah preferensi responden tersebut terhadap konten-konten pembelajaran yang bersifat kinestetik.

BAB V

KESIMPULAN DAN SARAN

51.1. Kesimpulan

Kesimpulan yang diperoleh pada penelitian ini adalah:

1. Aplikasi *App Usage Statistic Recorder* telah dikembangkan untuk proses pengumpulan data hasil kuesioner VARK dan data penggunaan aplikasi *smartphone*
2. Proses *query* pada obyek UsageStats dapat menghasilkan data durasi penggunaan aplikasi, sedangkan proses *query* pada obyek UsageEvents dapat menghasilkan data frekuensi penggunaan aplikasi.
3. Data durasi dan frekuensi penggunaan aplikasi *smartphone* memiliki korelasi yang signifikan terhadap model gaya belajar VARK.
4. Terdapat 3 skenario pengamatan yang dapat diterapkan untuk mengamati gaya belajar VARK berdasarkan penggunaan aplikasi *smartphone*.
5. Parameter-parameter *smartphone* yang layak untuk diamati pada skenario pengamatan bebas adalah durasi penggunaan aplikasi Adobe Lightroom CC, Android Calculator 2, Android MMS, DU-Recorder, MS Word, Mobile Legends, Pinterest, Samsusng Video Player, Hay day, serta frekuensi penggunaan WPS Office, Android Camera, dan Facebook.
6. Parameter-parameter *smartphone* yang layak untuk diamati pada skenario pengamatan berdasarkan kelas VARK dominan pengguna adalah durasi penggunaan Youtube terhadap aural, Microsoft PowerPoint terhadap kinestetik, serta frekuensi penggunaan aplikasi telpon Android terhadap aural, dan frekuensi penggunaan Google Maps terhadap kinestetik.

7. Parameter-parameter *smartphone* yang layak untuk diamati pada skenario pengamatan berdasarkan fungsi aplikasi adalah durasi penggunaan kalkulator, kamus, aplikasi-aplikasi edukasi, permainan simulasi, penyunting konten visual, serta frekuensi penggunaan aplikasi penampil informasi cuaca, dan aplikasi-aplikasi untuk komunikasi.

51.1. Saran

Berikut merupakan beberapa saran untuk penelitian selanjutnya:

1. Parameter-parameter *smartphone* yang layak digunakan untuk mengamati gaya belajar VARK pengguna sudah ditemukan, namun belum ditemukan faktor-faktor apa saja yang dapat berpengaruh terhadap hubungan dari tiap parameter terhadap VARK. Maka, penelitian selanjutnya dapat dilakukan untuk mengamati tiap parameter secara spesifik dan mendalam.
2. Perlu dilakukan penelitian lebih lanjut dengan jumlah responden yang lebih banyak agar lebih banyak pula aplikasi-aplikasi *smartphone* yang tercakup.
3. Agar dilakukan pengembangan layanan yang bekerja secara periodik sehingga data yang diperoleh lebih lengkap dan mendukung proses deteksi gaya belajar secara dinamis.

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 Institute 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. Heberts, 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] Laerd Statistic, "Spearman's Rank-Order Correlation," Lund Research Ltd, 2018. [Online]. Available: <https://statistics.laerd.com/statistical-guides/spearmans-rank-order-correlation-statistical-guide.php>. [Diakses 2 Juni 2018].

- [36] P. Sahoo, Probability and Mathematical Statistics, Louisville, Kentucky: University of Louisville, 2013.
- [37] statstutor, "Spearman's correlation," [Online]. Available:
<http://www.statstutor.ac.uk/resources/uploaded/spearmans.pdf>. [Diakses 2 Juni 2018].

LAMPIRAN