

PHOTO SHARING APPLICATION FOR GROUPS USING FLUTTER FRAMEWORK

Calvin Leonardi¹, Indra Maryati^{*2}

¹Informatika, Fakultas Sains dan Teknologi, Institut Sains dan Teknologi Terpadu Surabaya, Indonesia

²Sistem Informasi, Fakultas Teknologi Informasi, Universitas Ciputra Surabaya, Indonesia

Email: ¹penulis1calvin_agus@yahoo.co.id, ²indra.maryati@ciputra.ac.id

(Naskah masuk: 27 Mei 2022, Revisi : 30 Mei 2022, diterbitkan: 28 Juni 2022)

Abstract

Taking photos while traveling is certainly familiar today. Sharing the photos taken is also one of the mandatory parts to do, whether traveling alone, as a family, or in a group. Taking and sharing these photos is used for memories and even life satisfaction. In general, the activity of sharing photos that are captured using social media such as WhatsApp, LINE, Instagram, or others. The process of sharing photos on social media has limitations, where the photos that are shared must download first because there is no preview of the photos. This download process will use storage on the user's device. This is certainly not efficient if the photos sent are not in accordance with user context. The development of this photo sharing application uses the SDLC method starting from the analysis process to testing. This application provides a system where users can view photos shared by other users without download. This application will also backup photos on Firebase Storage as well as Google Photos. Image Labeling technology from Google ML Kit also helps to extract the information contained in the image and the extraction results will be used as a hashtag on the photo and can be used to search images using hashtags. The test was carried out on 10 testers and the application can help in sharing photos and there is no need to download to be able to view the shared photos. In addition, the use of image labeling from the Google ML Kit library is also very helpful in performing image searches.

Keywords: flutter, image labeling, ML Kit, photo sharing, google photos api.

APLIKASI PHOTO SHARING UNTUK GRUP DENGAN MENGGUNAKAN FLUTTER FRAMEWORK

Abstrak

Pengambilan foto saat melakukan perjalanan wisata tentunya tidak asing dilakukan masyarakat saat ini. Berbagi foto yang diambil tersebut juga menjadi salah satu bagian wajib yang dilakukan, baik melakukan perjalanan sendiri, keluarga, maupun suatu grup. Pengambilan dan berbagi foto tersebut digunakan untuk kenangan bahkan kepuasan hidup. Pada umumnya kegiatan membagikan foto-foto yang diabadikan dengan menggunakan media sosial seperti WhatsApp, LINE, Instagram, atau lainnya. Proses berbagi foto yang dilakukan pada media sosial tersebut memiliki keterbatasan, dimana foto yang dibagikan akan dilakukan proses unduh terlebih dahulu karena foto yang dikirimkan tidak dapat dilihat *preview*nya terlebih dahulu. Proses unduh ini akan menggunakan *storage* atau penyimpanan memori pada perangkat pengguna. Hal ini tentunya tidak efisien apalagi jika foto yang dikirimkan tidak sesuai dengan keinginan pengguna tersebut. Pengembangan aplikasi *photo sharing* ini menggunakan metode SDLC dimulai dari proses analisis hingga uji coba. Aplikasi ini menyediakan sistem dimana user dapat melihat foto-foto yang dibagikan oleh user lainnya tanpa perlu melakukan download untuk foto tersebut. Aplikasi ini juga akan melakukan backup foto-foto pada Firebase Storage dan juga Google Photos. Teknologi Image Labeling dari Google ML Kit juga membantu untuk mengekstraksi informasi yang ada pada gambar dan hasil dari ekstraksi tersebut akan dijadikan *hashtag* pada foto dan dapat digunakan untuk melakukan pencarian gambar dengan menggunakan *hashtag*. Pengujian dilakukan kepada 10 orang *tester* dan aplikasi dapat membantu dalam berbagi foto dan tidak perlu untuk melakukan proses unduh untuk dapat melihat foto-foto yang dibagikan tersebut. Selain itu penggunaan *image labeling* dari library Google MLKit juga sangat membantu dalam melakukan pencarian gambar.

Kata kunci: flutter, image labeling, ML Kit, photo sharing, google photos api.

1. PENDAHULUAN

Beberapa orang biasanya lebih suka menghabiskan waktu untuk pergi rekreasi, misalnya

seperti berwisata, makan bersama teman atau keluarga, dan aktivitas hiburan lainnya. Pada saat kita melakukan field trip atau perjalanan ke luar kota maupun ke luar negeri, tentunya orang-orang akan melihat-lihat dan memfoto tempat-tempat atau *spot* yang bagus untuk mengabadikan momen tersebut dengan menggunakan *smartphone* milik mereka masing-masing. Biasanya orang bepergian dengan keluarga, kerabat atau teman-teman mereka dan mereka biasanya akan membagikan foto-foto yang telah mereka ambil dengan orang-orang yang bepergian dengan mereka. Pengambilan foto yang penuh perhatian dapat meningkatkan kenikmatan dan ingatan akan pengalaman, membangkitkan suasana hati yang positif, dan meningkatkan kepuasan hidup [1].

Selain itu, penelitian [2] menemukan motif dibalik mengapa orang suka untuk mengunggah fotonya ke media sosial seperti Instagram, salah satu motifnya adalah motif karena (*because motives*) dimana motif ini muncul terkait dengan pengalamannya dalam album foto maupun berbagi foto di masa lampau. Sebelum teknologi berkembang pesat, kita dapat melihat foto dengan mencetaknya dan tidak dengan leluasa untuk dibagikan ke orang lain karena kendala biaya, waktu, tidak efisien, penyimpanan yang susah, dan lain sebagainya. Aplikasi komunikasi dan berbagi foto yang digunakan umumnya untuk orang tua adalah Whatsapp dan untuk anak muda umumnya adalah LINE. Grup WhatsApp adalah satu-satunya media komunikasi jaringan dan dibentuk dengan menambahkan nomor telepon orang ke grup itu. Sebaliknya, jejaring sosial lain didasarkan pada keanggotaan pengguna dan terutama berfokus pada pesan publik di mana pesan ini dikirim ke semua pengguna yang terhubung (yaitu pesan ini disebut Posting di Facebook dan Tweet di Twitter), dan bukan melalui grup pribadi [3].

Untuk dapat menyimpan foto yang telah dibagikan di grup, orang tersebut harus mengunduh foto yang telah dibagikan oleh temannya atau kerabatnya terlebih dahulu melalui aplikasi tempat mereka semua berbagi foto perjalanan mereka dan

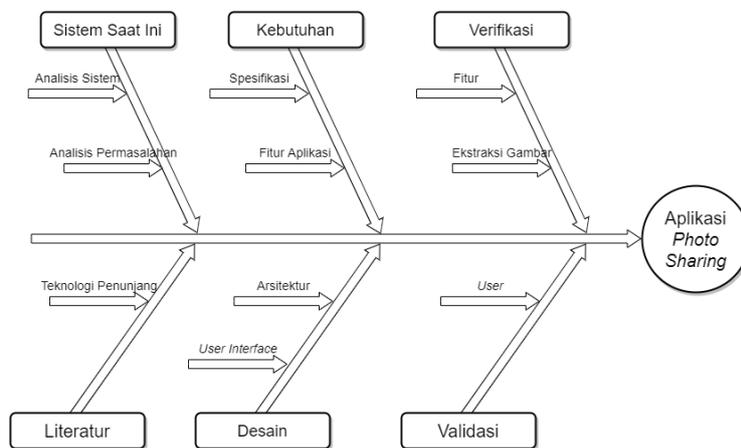
foto tersebut akan masuk ke dalam galeri pada *smartphone* mereka masing-masing. Dengan mengunduh foto-foto yang ada tentunya akan mengurangi *storage* yang ada pada *smartphone* sehingga biasanya *storage* tersebut akan penuh hanya karena berisikan banyak gambar dan juga video. Masalah penyimpanan ini tidak hanya terjadi pada individual, namun juga terjadi pada suatu bisnis fotografi. Hal ini tertuang pada penelitian [4] yang melakukan optimasi penyimpanan foto dengan menggunakan algoritma yang kemudian foto disimpan pada *website*. Dari penelitian tersebut dapat diambil peranan penting *storage* dalam perangkat *smartphone* yang digunakan oleh admin untuk mengirimkan gambar.

Dalam media sosial Instagram, pengguna sering menggunakan *hashtag* untuk mempermudah dalam melakukan pencarian. Dalam penelitian [5] mengatakan bahwa 55% dari partisipan dan pemilik gambar setuju bahwa *hashtag* yang disarankan dapat menjelaskan konten visual dalam sebuah gambar atau foto.

Untuk mengatasi masalah tersebut, dikembangkan sebuah sistem yang dapat membantu orang-orang yang bepergian dan mengabadikan momen yang ada. Fitur ini diintegrasikan dengan Google Photos sehingga pengguna dapat melihat langsung dan tidak perlu mengunduh dahulu foto yang diinginkan. Selain itu digunakan juga library Google ML Kit yang akan digunakan untuk pemberian *hashtag* secara otomatis, dimana disebut sebagai *image labeling* pada penelitian ini. Pemilihan flutter sebagai teknologi pengembangan aplikasi karena kemudahan yang ditawarkan, yaitu dari 1 *code* dapat diimplementasikan ke 2 *mobile platform* berbeda.

2. METODE PENELITIAN

Dalam penelitian ini menggunakan metodologi SDLC pada umumnya dimana diawali dengan proses analisis hingga uji coba. Gambar 1 adalah gambaran dari tahapan yang dilakukan pada penelitian ini.



Gambar 1. Tahapan Penelitian

Penelitian diawali dengan melakukan analisis dari sistem yang telah digunakan saat ini, mulai dari analisis sistem tersebut sampai dengan analisis pemasalahan pada sistem tersebut. Dari analisis akan didapatkan analisis kebutuhan, serta untuk menjawab kebutuhan tersebut diperlukan studi literatur dan teknologi penunjang yang dapat digunakan untuk mengembangkan aplikasi ini. Setelah proses analisis dilakukan proses desain mulai dari desain arsitektur sistem sampai desain tampilan atau *user interface* dari aplikasi *photo sharing*. Desain dilanjutkan dengan proses implementasi fitur-fitur dan sekaligus juga melakukan verifikasi dari fungsionalitas aplikasi. Validasi akan dilakukan dengan melakukan uji coba langsung ke user.

2.1. Analisis Sistem Berbagi Foto Saat Ini

Kebanyakan orang akan mengabadikan momen-momen pada saat rekreasi ataupun hobi mereka dengan menggunakan foto atau bisa juga dengan menggunakan video. Foto ataupun video tersebut biasanya diambil menggunakan perangkat yang mereka bawa seperti kamera digital ataupun smartphone yang mereka miliki. Foto-foto itu akan tersimpan pada memori pada perangkat tersebut, setelah itu biasanya orang-orang tersebut akan membagikan foto-foto hasil dari rekreasi mereka ataupun hobi yang mereka lakukan pada teman, keluarga, grup, ataupun melakukan *posting* pada akun mereka sendiri melalui media sosial yang mereka punyai seperti LINE, WhatsApp, atau Instagram. Menyadari pertumbuhan aplikasi perpesanan pribadi dan berbagi foto, Facebook mengakuisisi WhatsApp pada tahun 2014 dan Instagram pada tahun 2012 [6]. Jika pengguna mengirimkan foto dan mengunduhnya melalui aplikasi perpesanan pribadi, maka foto akan tersimpan pada *storage*.

2.2. Analisis Permasalahan

Pada analisis sistem berbagi foto saat ini, banyak orang akan melakukan *sharing* foto kepada keluarga, teman, grup dan lainnya melalui sosial media yang dimiliki. Sehingga pada umumnya akan memenuhi *storage* yang ada pada perangkat yang digunakan tersebut. Oleh karena itu timbul beberapa permasalahan yang umumnya terjadi, yaitu:

- Banyaknya foto ataupun video yang diambil pada saat berlibur menggunakan *smartphone* ataupun kamera yang bisa membuat memori penyimpanan (*storage*) pada perangkat menjadi penuh.
- Dengan adanya sosial media yang banyak sekali, akan mempersulit pengguna untuk membagikan foto dan video tersebut. Karena setiap orang bisa saja menggunakan sosial media yang berbeda-beda.

- Karena banyaknya foto yang dikirim akan membuat orang-orang yang menerima kesulitan untuk menemukan fotonya sendiri.

2.3. Spesifikasi Kebutuhan

Pada spesifikasi kebutuhan akan dijelaskan secara terperinci mengenai analisis kebutuhan sistem yang akan menjawab dari semua permasalahan yang ada pada subbab analisis kebutuhan. Berikut ini adalah beberapa fitur dari sistem yang dibuat yang bisa menjadi solusi atas permasalahan sebelumnya:

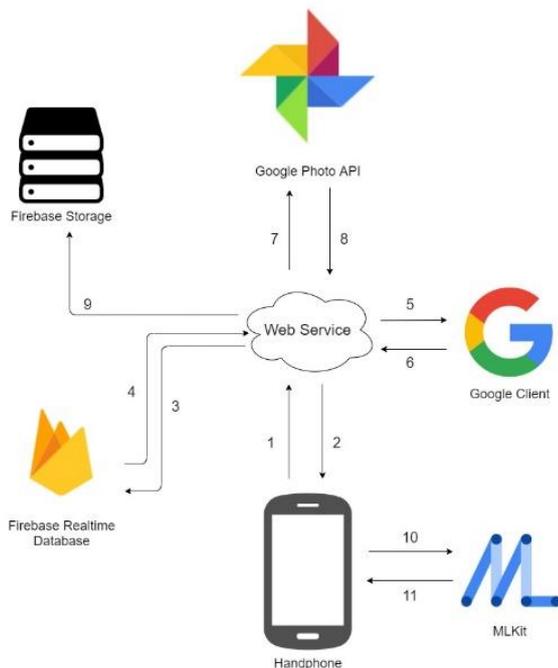
- Login dengan menggunakan akun Google akan memberikan kemudahan pada pengguna sehingga hanya perlu memilih akun mana yang akan digunakan untuk login dan tidak perlu untuk melakukan login secara manual seperti memasukan nama, alamat email, nomor handphone, dan lainnya satu persatu.
- Fitur pencarian dengan *hashtag* dimana akan berguna dan memudahkan jika orang lain ingin menemukan foto.
- Adanya fitur *backup* foto pada *cloud* yang akan sangat berguna untuk pengguna dimana foto yang diunggah nantinya akan *backup* dan pengguna dapat menghapus foto yang ada pada *smartphone* sehingga dapat mengurangi pemakaian memori yang ada pada perangkat yang digunakan.

3. HASIL DAN PEMBAHASAN

Arsitektur dirancang guna mempermudah pembacaan, perancangan dan pembentukan sistem. Beberapa teknologi yang digunakan dalam pengembangan aplikasi ini, yaitu Google Client, Google Photos API, Flutter, Firebase Storage, Firebase Realtime Database, dan Google MLKit. Google Client akan digunakan untuk proses otentikasi dan dalam pemberian *token* album. Google Photos API digunakan untuk pembuatan album foto dan pencarian yang lebih mudah. Flutter merupakan open-source UI software untuk mengembangkan aplikasi *cross-platform* [7]. Flutter dikembangkan dengan menggunakan bahasa pemrograman Dart [8][9]. Firebase Storage[10] dan Firebase Realtime Database [11] digunakan untuk menyimpan data pengguna secara *realtime* dan dapat diakses dimana saja. Google ML Kit merupakan *library* yang digunakan untuk melakukan *image labeling*. Selain *image labeling*, Google ML Kit juga dapat melakukan OCR [12] yang memperoleh hasil cukup baik, *Translate*, *object detection*, dll. *Webservice* dibuat dengan menggunakan RESTful dan ditempatkan pada *web server* pribadi. Penggunaan RESTful lebih mudah dibandingkan dengan menggunakan SOAP [13]. Gambar 2 adalah rancangan sistem yang dibuat dan menjelaskan keterhubungan antar teknologi yang digunakan.

Gambar 2 merupakan arsitektur dari sistem yang dibuat. pada arsitektur ini terdapat beberapa

API yang akan mendukung dari sistem yang akan dibuat yaitu: Google Client, Firebase Realtime Database, Firebase Storage dan Google Photos API. Berikut ini adalah penjelasan dari masing-masing nomor yang diberikan.

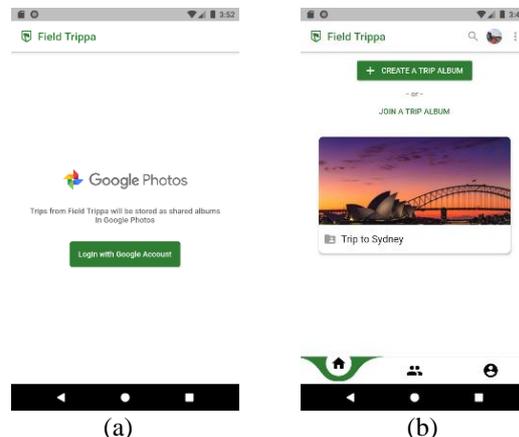


Gambar 2. Arsitektur Diagram

1. Aplikasi yang ada pada *handphone* user dapat melakukan *request* ke *webservice* untuk melakukan register, login, menampilkan data, dll.
2. *Webservice* akan mengembalikan permintaan dari aplikasi android.
3. *Webservice* akan mengakses *database* untuk memasukkan dan mengubah data yang ada berdasarkan permintaan user.
4. *Database* akan mengembalikan data yang diminta.
5. *Webservice* akan melakukan *request* kepada Google Client untuk verifikasi login menggunakan akun Google.
6. Google Client akan memberikan informasi data pengguna sesuai dengan *token* yang telah dikirimkan.
7. *Webservice* akan mengakses Google Photos untuk mengakses fitur seperti membuat album, memasukkan foto, dll.
8. Google Photos akan mengembalikan semua album beserta foto-foto yang ada didalamnya untuk dapat diakses oleh user.
9. *Webservice* akan mengakses Firebase Storage untuk melakukan *backup* foto.
10. Aplikasi Android mengirim gambar pada Google ML Kit untuk diproses sesuai permintaan. Pada sistem ini aplikasi Android akan melakukan *request Image Labeling* [14].

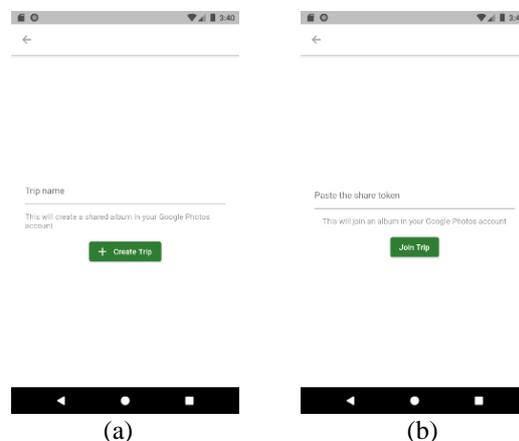
11. Google ML Kit akan mengembalikan data berupa sekumpulan *text* pada aplikasi Android.

Terkait dengan desain tampilan aplikasi akan dijelaskan mulai dari tampilan *login* sampai dengan melakukan *sharing* album foto ke user lain. Desain dari aplikasi dibuat dengan menggunakan komponen yang disediakan oleh Flutter.



Gambar 3. Tampilan Login dan Dashboard

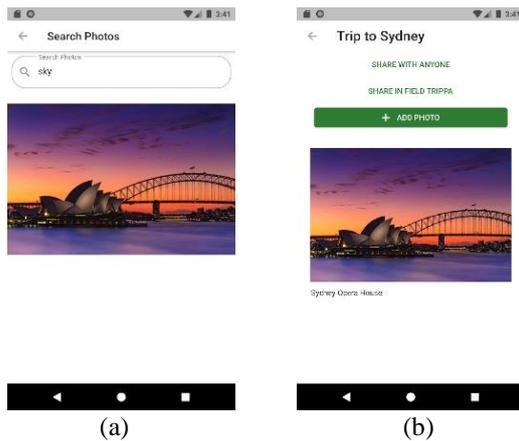
Gambar 3(a) merupakan merupakan tampilan awal yang akan didapat ketika aplikasi pertama kali terunduh dan terinstall pada perangkat, halaman tersebut adalah halaman *login* pada aplikasi. Pada halaman ini pengguna dapat langsung login dengan menggunakan akun Google yang dimiliki. Penggunaan akun Google ini agar mempermudah pengguna untuk melakukan *login*. Setelah melakukan *login*, pengguna akan diarahkan kedalam halaman *dashboard* seperti pada gambar 3(b). Pada halaman ini akan terdapat album yang sudah dibuat oleh pengguna, lalu ada tombol untuk membuat album baru, lalu tulisan "Join A Trip Album" digunakan untuk *join* album orang lain dan juga ada tombol *search* foto pada appbar. Titik 3 pada appbar dapat digunakan jika pengguna ingin *disconnect* dari aplikasi dan menggunakan akun Google lainnya.



Gambar 4. Tampilan Buat Album dan Join Album

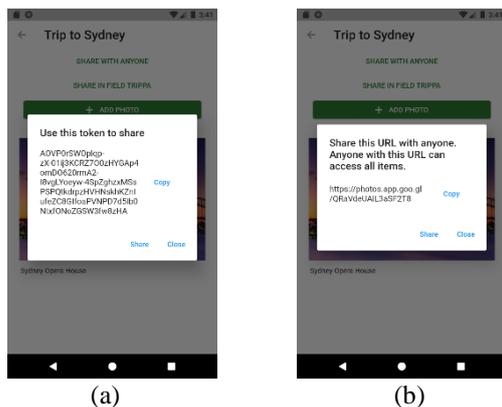
Jika tombol buat album baru yang ada pada gambar 3(b) ditekan maka pengguna akan diarahkan

pada halaman buat album baru seperti pada gambar 4(a). Pada halaman ini pengguna wajib mengisi nama album yang akan dibuat. Jika tombol *join* album yang ada pada gambar 3(b) ditekan maka pengguna akan diarahkan pada halaman *join* album seperti pada gambar 4(b). Pada halaman ini pengguna dapat memasukkan *token*, yang bisa didapatkan jika ada pengguna lainnya yang ingin *sharing* album yang dibuat dengan pengguna aplikasi lainnya. *Token* bisa didapatkan dengan menekan “Share in Field Trippa” pada halaman detail album.



Gambar 5. Tampilan Search Foto dan Detail Album

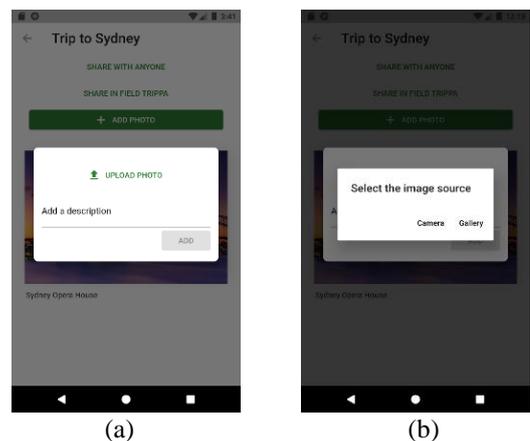
Gambar 5(a) merupakan tampilan pada halaman *search* foto. Pada halaman ini pengguna dapat melakukan pencarian foto yang ada pada seluruh album dengan mengisi *hashtag* yang ingin dicari. *Hashtag* yang ada pada foto dihasilkan dari fitur *Image Labeling* dari ML Kit. Gambar 5(b) adalah halaman detail album, untuk dapat mengakses halaman ini pengguna dapat menekan salah satu album yang ada pada *dashboard* pengguna. Pada halaman ini pengguna dapat melihat semua foto yang sudah pernah diunggah ke dalam album tersebut dan terdapat tombol tambah foto yang dapat digunakan oleh pengguna untuk menambahkan foto baru ke dalam album tersebut. Pengguna juga dapat *sharing* album yang dibuat dengan menekan tulisan *share* yang ada di atas tombol tambah foto.



Gambar 6. Tampilan Token Share dan URL Share

Gambar 6(a) merupakan sebuah *dialog* yang akan muncul jika tulisan “Share in Field Trippa” ditekan. *Dialog* ini akan menampilkan sebuah *token* yang dapat digunakan untuk mengundang pengguna aplikasi lainnya untuk mendapatkan akses pada album yang dibagikan tersebut. Pada *dialog* ini akan disediakan fungsi untuk melakukan *copy token* dan *share* untuk memudahkan pengguna untuk melakukan *sharing*. Gambar 6(b) merupakan sebuah *dialog* yang akan muncul jika tulisan “Share With Anyone” ditekan. Fungsi pada *dialog* ini hampir sama dengan gambar 6(a), yang membedakan adalah *dialog* yang muncul akan memberikan url untuk dibagikan dan bukan sebuah *token*. *Url* ini dapat digunakan oleh semua orang walaupun orang tersebut tidak memiliki aplikasi yang dibuat ini karena aplikasi yang dibuat dihubungkan juga dengan Google Photos menggunakan Google Photos API [15].

Gambar 7(a) merupakan *dialog* yang akan muncul jika pengguna menekan tombol “Add Photo” pada halaman detail album. Pada *dialog* ini akan terdapat tombol “Upload Photo” yang dapat digunakan pengguna untuk memilih foto yang ingin diunggah ke dalam album. Setelah menekan tombol “Upload Photo” pengguna akan diberi pilihan seperti pada gambar 7(b). Pengguna dapat memilih ingin memasukkan foto dari kamera atau dari galeri yang ada di perangkat pengguna. Setelah memilih foto pengguna dapat memasukkan deskripsi foto yang diinginkan, jika sudah diisi maka pengguna dapat mengupload foto tersebut dengan menekan tombol “Add” yang ada dibawah deskripsi.



Gambar 7. Tampilan Add Photo dan Image Source

Pengujian Perangkat Lunak sangat penting dan merupakan bagian yang intensif dari siklus hidup pengembangan perangkat lunak [16]. Bahkan ada juga proses pengujian perangkat lunak yang menggunakan robot[17]. Google sendiri tentunya telah melakukan pengujian terhadap library Google ML Kit yang dibuatnya. Seperti yang diutarakan oleh [18], agar AI bisa berjalan dengan baik perlu dilakukan *training* dan *testing* agar mesin tersebut

semakin pintar. Dan mereka menggunakan TensorFlow untuk mengujinya.

Proses verifikasi dilakukan oleh *developer* untuk mengetahui apakah fitur sistem sudah berfungsi sesuai dengan fungsionalitasnya. Sedangkan proses validasi dilakukan oleh *developer* ke *user* atau pengguna untuk mengetahui apakah fitur yang ditawarkan sistem sudah sesuai dengan keinginan pengguna. Untuk proses verifikasi dilakukan pengecekan terhadap semua fitur yang telah dibuat dan mendapati hasil sekitar 98% aplikasi telah berjalan dengan baik dan sesuai fungsionalitasnya. Terdapat beberapa kegagalan dalam melakukan proses ekstraksi informasi berdasarkan gambar, atau dalam melakukan *image labeling*. Kegagalan disebabkan karena foto yang dimasukkan ke album foto memiliki kualitas yang kurang baik, sehingga Google ML Kit tidak dapat melakukan ekstraksi informasi dengan baik. Untuk uji coba *image labeling*, uji coba pertama dilakukan dengan menggunakan gambar dari Opera House Sydney.



Gambar 8. Uji Coba Image Labeling Pertama

Label	Confidence
Bridge	0.913040
Sky	0.882152
Sunset	0.879520
Dog	0.649014
Metal	0.636565
Bird	0.553215
River	0.537125

Gambar 8 merupakan hasil deteksi yang dilakukan pada saat uji coba. Dapat dilihat bahwa Google ML Kit berhasil mengekstrak beberapa informasi yang ada pada gambar. Pada gambar ini terdapat 7 informasi yang diekstrak oleh fitur Image Labeling pada Google ML Kit. Pada gambar ini dapat dilihat pada tabel bahwa semakin tinggi nilai *confidence* maka informasi yang didapat akan semakin sesuai. Pada label ke-4 terdapat label *Dog*, sedangkan pada gambar tidak terdapat anjing sama sekali, maka dari itu nilai *confidence* nya rendah. Sedangkan untuk *Bridge*, *Sky*, dan *Sunset* memiliki nilai *confidence* yang cenderung tinggi karena informasi tersebut sesuai pada gambar. Sehingga label yang akan diambil adalah *Bridge*, *Sky*, dan *Sunset*.



Gambar 9. Uji Coba Image Labeling Kedua

Label	Confidence
Sky	0.919049
Boat	0.881433
Vehicle	0.850131
River	0.823894
Vacation	0.771083
Mountain	0.746498
Sunset	0.745132
Leisure	0.545670

Gambar 9 merupakan hasil uji coba yang ke-2. Pada hasil uji coba yang ke-2 Google ML Kit memberikan beberapa label dengan nilai *confidence* yang cenderung tinggi tetapi juga ada label yang memiliki nilai *confidence* cukup rendah seperti

Leisure. Ada 4 label yang memiliki nilai *confidence* yang bagus seperti *Sky*, *Boat*, *Vehicle*, dan *River*. Sehingga label yang akan diambil adalah *Sky*, *Boat*, *Vehicle*, dan *River*.

Uji coba terhadap aplikasi secara keseluruhan juga dilakukan ke user secara langsung. Uji coba dilakukan ke 10 orang dengan mencoba hasil aplikasi yang sudah dibuat. Dalam melakukan uji coba, *tester* akan mengunduh *file* APK terlebih dahulu, kemudian menginstall pada *smartphone* masing-masing. Respons dari *tester* akan ditampung dalam kuesioner yang dibuat dengan menggunakan Google Form. Tabel 1 merupakan hasil kuesioner dari 3 pertanyaan yang diajukan dengan 4 presentasi jawaban

Tabel 1. Hasil Kuesioner

Pertanyaan	Poin (dalam %)				
	0	1	2	3	4
Apakah tampilan aplikasi nyaman untuk dilihat? (Segi warna, jenis komponen, ukuran komponen, letak komponen)	0	10	0	50	40
Apakah aplikasi dapat membantu untuk <i>sharing</i> foto?	0	0	0	50	50
Apakah foto dapat dilihat tanpa perlu diunduh sebelumnya?	0	0	0	40	60

Tabel 1 adalah tabel hasil uji coba yang telah dilakukan oleh *tester*. Dalam perhitungan prosentase menggunakan persamaan (1).

$$\% \text{poin} = \frac{\sum \text{Answer}}{N} \cdot 100\% \quad (1)$$

Pada persamaan (1) $\sum \text{Answer}$ merupakan jumlah dari *tester* yang menjawab dengan poin tersebut. N merupakan jumlah *tester*, dimana dalam penelitian ini berjumlah 10. Pada pertanyaan 1, *tester* diminta untuk menilai segi kenyamanan aplikasi yang telah dibuat. Dari 10 *tester*, terdapat 5 orang yang memberikan 3 poin dan 4 orang memberikan 3 poin. Pada pertanyaan 2, *tester* diminta untuk menilai segi kegunaan aplikasi yang telah dibuat, terdapat 5 orang memberikan 3 poin dan 5 orang memberikan 4 poin. Pada pertanyaan 3, *tester* diminta untuk menilai dari segi fungsionalitas salah satu fitur yaitu unduh foto. Terdapat 4 orang menjawab poin 3 dan 6 orang menjawab poin 4.

Dari hasil penilaian dari tabel dan dilihat untuk poin 3 dan 4 khususnya, dapat disimpulkan bahwa aplikasi nyaman untuk dilihat dengan perolehan nilai 90% dan 10% menyatakan kurang cukup. Pada pertanyaan kedua dapat disimpulkan bahwa aplikasi dapat membantu user untuk melakukan *sharing* foto dengan perolehan nilai 100%. Pada pertanyaan ketiga dapat disimpulkan bahwa fitur foto dapat dilihat tanpa perlu diunduh sebelumnya dapat membantu user dengan perolehan nilai 100%.

4. KESIMPULAN

Aplikasi foto *sharing* dikembangkan memiliki fitur pembuatan album foto, berbagi album foto,

dapat memberikan deskripsi dengan *hashtag* secara otomatis, dapat melakukan pencarian, dapat mengundang teman untuk menggunakan aplikasi ini, dan yang terpenting adalah dapat melihat foto tanpa perlu melakukan unduh foto terlebih dahulu.

Dari uji coba yang telah dilakukan kepada 10 orang *tester*, aplikasi photo sharing yang dikembangkan dapat membantu dalam melakukan *sharing* foto atau berbagi foto dan tidak perlu untuk melakukan proses unduh untuk dapat melihat foto-foto yang dibagikan tersebut. Selain itu penggunaan *image labeling* dari library Google MLKit juga sangat membantu dalam melakukan pencarian gambar, dimana hasil dari image labeling ditambahkan pada deskripsi foto sehingga dapat mempermudah pencarian. Pemilihan *label* hasil dari pemrosesan Google ML Kit, sebaiknya dipilih yang memiliki nilai *confidence* diatas 80%.

DAFTAR PUSTAKA

- [1] K. Diehl and G. Zauberman, "Capturing life or missing it: How mindful photo-taking can affect experiences," *Curr. Opin. Psychol.*, vol. 46, p. 101334, 2022, doi: 10.1016/j.copsy.2022.101334.
- [2] A. Febrina, "MOTIF ORANG TUA MENGUNGGAH FOTO ANAK DI INSTAGRAM (Studi Fenomenologi Terhadap Orang Tua di Jabodetabek)," *J. Abdi Ilmu*, vol. 12, no. 1, pp. 55–65, 2019, [Online]. Available: <https://journal.pancabudi.ac.id/index.php/abdiilmu/article/view/529>.
- [3] A. Rosenfeld, S. Sina, D. Sarne, O. Avidov, and S. Kraus, "A Study of WhatsApp Usage Patterns and Prediction Models without Message Content," pp. 1–24, 2018, [Online]. Available: <http://arxiv.org/abs/1802.03393>.
- [4] A. Bahrudin, P. Permata, and J. Jupriyadi, "Optimasi Arsip Penyimpanan Dokumen Foto Menggunakan Algoritma Kompresi Deflate (Studi Kasus :Studio Muezzart)," *J. Ilm. Infrastruktur Teknol. Inf.*, vol. 1, no. 2, pp. 14–18, 2020, doi: 10.33365/jiiti.v1i2.582.
- [5] S. Giannoulakis, N. Tsapatsoulis, S. Giannoulakis, N. Tsapatsoulis, I. Hashtags, and A. Metadata, "Instagram Hashtags as Image Annotation Metadata," *11th IFIP Int. Conf. Artif. Intell. Appl. Innov. (AIAI 2015)*, pp. 206–220, 2016, [Online]. Available: <https://hal.inria.fr/hal-01385356>.
- [6] B. Franklin and L. Canter, *Digital Journalism Studies: The Key Concepts*. Taylor & Francis, 2019.
- [7] Google, "Flutter Documentation," *Flutter.Dev*, 2021. <https://flutter.dev/docs> (accessed Mar. 14, 2022).
- [8] G. Bracha, *The Dart Programming Language*. Addison-Wesley, 2016.
- [9] M. Belchin and P. Juberias, *Web Programming with Dart*. Apress, 2015.
- [10] Google, "Firebase Documentation," *Firebase.Google.Com*, 2021. <https://firebase.google.com/docs/flutter/setu?p?platform=android> (accessed Mar. 12, 2022).
- [11] A. K. S, *Mastering Firebase for Android Development: Build real-time, scalable, and cloud-enabled Android apps with Firebase*. Packt Publishing, 2018.
- [12] I. Maryati, "Website Perpustakaan 'Library HUB' dengan Pencarian Buku Berdasarkan Gambar Menggunakan Google MLKit," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 8, no. 4, pp. 1821–1831, Dec. 2021, doi: 10.35957/jatisi.v8i4.1269.
- [13] L. Richardson, M. Amundsen, and S. Ruby, *RESTful Web APIs: Services for a Changing World*. O'Reilly Media, 2013.
- [14] Google, "ML Kit Recognize Text." <https://developers.google.com/ml-kit/vision/text-recognition> (accessed May 09, 2021).
- [15] Google, "Google Photos API Documentation." <https://developers.google.com/photos> (accessed Mar. 10, 2022).
- [16] J. Ahmad, A. ul Hasan, T. Naqvi, and T. Mubeen, "A Review on Software Testing and Its Methodology," *Manag. J. Softw. Eng.*, vol. 13, no. 1, pp. 32–38, 2019, doi: 10.26634/jse.13.3.15515.
- [17] R. Marselis, "Testing in the Digital Age," in *The Future of Software Quality Assurance*, S. Goericke, Ed. Cham: Springer International Publishing, 2020, pp. 77–92.
- [18] L. Moroney, *AI and Machine Learning for On-Device Development*. O'Reilly Media, 2021.