



**PENGEMBANGAN APLIKASI EDUKASI PENGELOLAAN
SAMPAH UNTUK ANAK SEKOLAH DASAR BERBASIS *MOBILE*
DENGAN TEKNOLOGI *MACHINE LEARNING***

Ardian Pramudya Alphita¹, Pratyaksa Ocsa Nugraha Saian²

^{1,2}Program Studi Teknik Informatika, Fakultas Teknologi Informasi
Universitas Kristen Satya Wacana
Jl. Dr. O. Notohamidjodjo, Blotongan, Salatiga, Jawa Tengah 50715

Email : 672018150@student.uksw.edu¹, pratyaksa.ocsa@uksw.edu²

Riwayat artikel:

Submitted: 04-11-2022

Revised: 27-11-2022

Published: 04-02-2023

Abstrak – Pengelolaan sampah yang buruk telah menjadi permasalahan yang masih dialami di seluruh dunia, tak terkecuali Indonesia. Penedukasian mengenai pengelolaan sampah perlu ditingkatkan terutama pada anak-anak dengan memanfaatkan perkembangan teknologi yang ada seperti teknologi *smartphone*. Berdasarkan masalah tersebut, dikembangkanlah aplikasi edukasi mengenai pengelolaan sampah yang ditujukan untuk anak Sekolah Dasar dengan memanfaatkan teknologi Android dan *machine learning*. Dengan tingkat akurasi sebesar 90% pada *training* dan *testing dataset*, pemanfaatan teknologi *machine learning* ini akan efektif untuk membantu anak-anak dalam mendeteksi sampah yang ditemukan ketika bereksplorasi. Penelitian ini menggunakan metode Waterfall yang terdiri dari 5 tahapan yaitu *requirement, design, implementation, testing* dan *maintenance*. Dengan menggunakan metode pengujian Black Box dan Beta Test, hasil pengujian yang didapatkan pada pengujian Black Box telah memenuhi semua skenario yang ada, dan melalui Beta Test mendapatkan respons positif serta beberapa saran dari calon pengguna untuk pengembangan aplikasi ke depannya. Dengan hasil yang didapatkan, aplikasi edukasi mengenai pengelolaan sampah dapat menjadi sarana untuk membantu anak-anak dalam memahami betapa pentingnya pengelolaan sampah melalui media pembelajaran interaktif.

Kata Kunci – Android, Edukasi, Kecerdasan Buatan, Media Belajar, Sampah .

Abstract – *Poor waste management has become a problem that is still experienced throughout the world, including Indonesia. Education about waste management needs to be improved, especially for children by utilizing existing technological developments such as smartphone technology. Based on this problem, an educational application about waste management was developed aimed at elementary school children by utilizing Android technology and machine learning. With an accuracy rate of 90% in training and testing datasets, the use of machine learning technology will be effective in helping children detect garbage found when exploring. This research uses the Waterfall method which consists of 5 phases, namely requirements, design, implementation, testing, and maintenance. By using the Black Box and Beta*

Test methods, the test results obtained in the Black Box test have met all the existing scenarios, and through the Beta Test got a positive response as well as some suggestions from potential users for future application development. With the results obtained, educational applications regarding waste management can be a means to help children understand the importance of waste management through interactive learning media.

Keywords – *Android, Education, Learning Media, Machine Learning, Waste.*

I. PENDAHULUAN

Sampah menjadi hal yang tidak bisa dihindari dalam kehidupan sehari-hari. Setiap orang, setiap rumah, dan setiap industri telah menghasilkan sampah yang berbeda-beda setiap harinya. Pengelolaan sampah telah menjadi permasalahan yang dialami oleh setiap negara tak terkecuali dengan Indonesia. Berdasarkan data yang telah dikeluarkan oleh Kementerian Lingkungan Hidup dan Kehutanan (KLHK) pada *website* resminya, menyebutkan bahwa pada tahun 2020 Indonesia telah menghasilkan sedikitnya 33 juta ton timbunan sampah. Dari seluruh sampah tersebut sebesar 40% sampah di Indonesia tidak dikelola sama sekali yang berarti telah mencemari lingkungan. Dari banyaknya sampah yang ditimbun, sampah yang berasal dari sisa makanan dan yang berasal dari rumah tangga adalah produsen terbanyak [1].

Terdapat banyak faktor yang menyebabkan pengelolaan sampah dalam suatu daerah atau bahkan negara itu buruk. Selain karena banyaknya jumlah sampah yang dihasilkan, kesadaran mengenai penanganan sampah secara individu juga masih kurang. Masih banyak orang-orang yang tidak melakukan pemilahan sampah secara individu dan bahkan tidak mematuhi aturan dengan membuang sampah secara sembarangan [2]. Hal seperti inilah yang perlu dibenahi dari diri setiap orang sejak dini agar nantinya sudah menjadi suatu kebiasaan dan mengubah pemikiran skeptis mengenai pengelolaan sampah. Melihat kondisi ini, salah satu cara untuk membantu mengubah pemikiran masyarakat adalah dengan mengedukasi anak-anak sejak dini mengenai pengelolaan sampah.

Saat ini semakin banyak orang dari berbagai kalangan yang menggunakan *smartphone* untuk mendukung kegiatan sehari-hari. Orang-orang dapat berkirim pesan, melakukan transaksi belanja, hingga memesan ojek hanya dengan *smartphone*. Perkembangan internet yang semakin cepat pun juga mempengaruhi jumlah pengguna *smartphone*. Berdasarkan survei yang dilakukan oleh Asosiasi Penyelenggara Jasa Internet Indonesia (APJII), pengguna internet di pertengahan tahun 2020 sebanyak 196,71 juta jiwa dimana mayoritas pengguna internet menggunakan *smartphone* [3].

Dengan banyaknya pengguna *smartphone* dewasa ini, dapat menjadi sarana untuk membantu masyarakat dalam mengelola sampah di kegiatan sehari-hari melalui aplikasi yang disediakan. *Smartphone* sendiri menggunakan beberapa jenis sistem operasi, salah satunya adalah *Android*. Pada tahun 2018, pengguna *smartphone* dengan sistem operasi *Android* telah meningkat sebanyak 73% [4]. Dengan banyaknya pengguna sistem operasi *Android*, maka memiliki potensi akan lebih banyak layanan aplikasi yang dikembangkan untuk *Android* untuk menyesuaikan dengan pangsa pasar [5].

Machine learning memiliki peran untuk mengenali, mengidentifikasi, ataupun memprediksi data tertentu dengan data histori atau data yang telah dipelajari sebelumnya. *Machine learning* digunakan untuk membuat program yang bisa belajar dari data, dan

machine learning juga dirancang untuk mampu belajar sendiri [6]. *Machine learning* dapat diimplementasikan dalam teknologi yang digunakan dalam kehidupan sehari-hari, seperti pendeteksi wajah, asisten digital, hingga mobil pintar. Dalam riset ini, teknologi *machine learning* digunakan untuk membantu anak-anak mendeteksi sampah dengan menggunakan gambar atau foto jenis-jenis sampah yang biasa terdapat di lingkungan sekitarnya. Tersedianya dataset yang dapat diakses dengan bebas (*open source*) akan membantu proses pengembangan aplikasi ini. *Dataset* untuk sumber data diambil dari website *Kaggle* yang kemudian dimodifikasi dan dilatih kembali menggunakan metode *transfer learning* yang disediakan oleh *library* Tensorflow.

Oleh karena itu dibuatlah suatu aplikasi mengenai edukasi pengelolaan sampah dengan harapan dapat mengajarkan mengenai pengelolaan sampah kepada anak-anak pada usia dini, terkhususnya anak-anak yang berada di jenjang Sekolah Dasar. Dengan diajarkan melalui media interaktif, anak-anak nantinya dapat dengan mudah memahami dan menerapkan di dalam kehidupan sehari-hari [7].

II. KAJIAN PUSTAKA

Pada penelitian dengan judul “*Designing Environment Care Adventure Game Based on Android Using Construct 2*”, menuliskan bahwa *ICT-based learning* atau pembelajaran berbasis Teknologi Informasi dan Komputer (TIK) merupakan pembelajaran yang interaktif, menarik, mandiri dengan bantuan teknologi. Perkembangan teknologi menyebabkan munculnya banyak variasi media belajar. Terdapat banyak hal yang dapat digunakan untuk melaksanakan *ICT-based learning*, seperti tutorial, simulasi, dan bahkan gim [4].

Daulana et al. [8] menjelaskan bahwa melalui media interaktif seperti gim dapat menjadi media untuk menyampaikan informasi dan sosialisasi kepada masyarakat. Berdasarkan hasil penelitian yang telah dilakukan, sebanyak 77% responden menyebutkan bahwa mereka dapat menghabiskan waktu sekitar 4 sampai 6 jam untuk bermain game. Dengan begitu, perlu untuk memiliki upaya yang lebih dalam membentuk suatu gaya baru sebagai media penyuluhan kepada masyarakat, kemudian dibangun suatu aplikasi ataupun gim edukatif yang diharapkan dapat memperkenalkan dan mengajarkan masyarakat untuk lebih peduli dengan lingkungan sekitar.

Pada penelitian dengan judul “*Development of Sorting Waste Game Android Based for Early Childhood in Environmental Education*” menuliskan bahwa edukasi mengenai lingkungan perlu diberikan sejak usia dini. Anak-anak pada usia dini merupakan anak-anak yang sedang berada dalam masa pertumbuhan dan perkembangan yang unik. Menurut Rahmawati et al, belajar dengan menggunakan media interaktif seperti gim dapat memberikan suasana yang lebih santai ketika anak sedang belajar karena materi pembelajaran disajikan dalam bentuk suatu permainan [9].

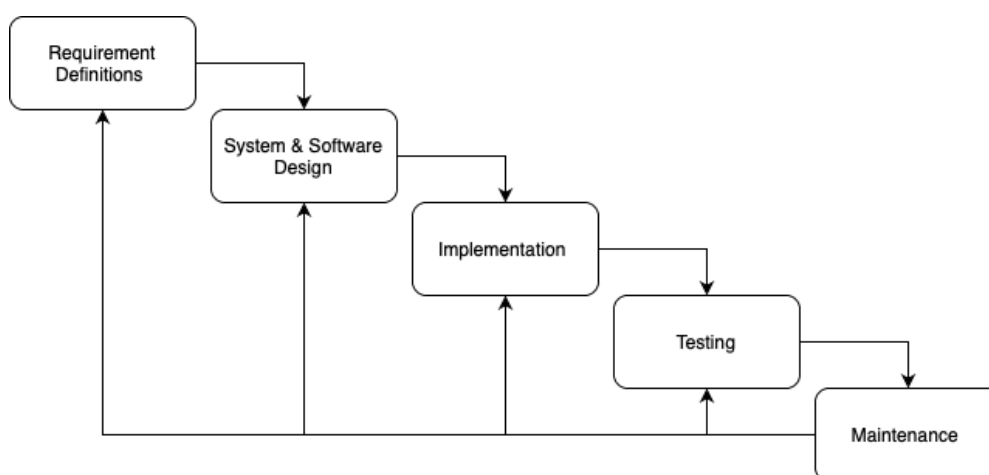
Berdasarkan jurnal hasil *review paper* dengan judul “Pemanfaatan Machine Learning Dalam Berbagai Bidang: Review Paper” yang ditulis oleh Ahmad Roihan et al menjelaskan bahwa kecerdasan buatan merupakan salah satu bidang dalam ilmu komputer yang ditujukan pada pembuatan perangkat lunak dan perangkat keras yang dapat berpikir atau menentukan keputusan seperti manusia. Terdapat banyak bidang yang telah menggunakan sistem kecerdasan buatan untuk membantu menyelesaikan masalah yang dihadapi seperti mengenai robotika, medis, keuangan, teknik dan gim [10].

Kaitan mengenai *machine learning* dengan penelitian mengenai pengembangan aplikasi edukasi mengenai pengelolaan sampah adalah dengan memanfaatkan teknologi *machine learning* yang ada, maka dapat mengembangkan media belajar interaktif yang lebih menarik. Dengan adanya teknologi *machine learning* dalam aplikasi edukasi mengenai pengelolaan sampah, diharapkan pengguna aplikasi dapat bereksplorasi secara langsung di lingkungan sekitar untuk belajar hal baru mengenai jenis-jenis sampah yang berada di sekitar pengguna.

Android merupakan suatu sistem operasi untuk perangkat mobile yang bersifat *open-source* berbasis Linux. Android awalnya dikembangkan oleh Android, Inc., namun akhirnya dibeli oleh Google pada tahun 2005 dan dirilis secara resmi pada tahun 2007. Antarmuka pengguna Android umumnya berupa manipulasi langsung, menggunakan gerakan sentuh yang serupa dengan tindakan nyata, seperti menggeser dan mengetuk untuk memanipulasi objek yang berada di layar [11].

III. METODE PENELITIAN

Berdasarkan tujuan-tujuan yang telah dipaparkan sebelumnya, berikut merupakan beberapa penguraian dan langkah-langkah penelitian yang akan dilakukan. Proses penelitian ini akan dilakukan dengan metode System Development Life Cycle (SDLC) Waterfall, dengan urutan seperti pada diagram berikut [12]:



Gambar 1. Tahapan dalam metode *Waterfall* [13]

Pada tahapan *requirement definitions*, akan dilakukan analisa mengenai semua kebutuhan yang akan digunakan dalam pengembangan aplikasi. Kebutuhan yang ditekankan dalam pengembangan aplikasi edukasi pengelolaan sampah ini adalah *data sample* berupa foto-foto sampah di lingkungan sekitar. *Data sample* akan digunakan dalam pembuatan model yang nantinya akan digunakan untuk *machine learning*. Selain *data sample* juga dibutuhkan materi edukasi mengenai pengelolaan sampah untuk ditampilkan di dalam aplikasi.

System & Software Design adalah tahap dimana desain disesuaikan dengan kebutuhan aplikasi yang akan dibuat terkait *database*, *architecture*, dan *user interface*. Penggunaan diagram seperti *Unified Modeling Language* (UML) juga dapat digunakan untuk menjelaskan dengan lebih terperinci dalam rancangan pembuatan aplikasi.

Implementaion adalah tahap dimana aplikasi akan dikembangkan pada *platform* Android menggunakan bahasa Kotlin. Di dalam aplikasi akan menerapkan model yang dibuat berdasarkan *data sample* yang telah dikumpulkan, kemudian dilakukan . Aplikasi dapat mendeteksi sampah dengan menerima masukan dari hasil pengambilan gambar pada *smartphone*. Selain memiliki fitur untuk mendeteksi sampah, aplikasi juga akan memiliki fitur edukasi berupa kuis sederhana untuk memberikan edukasi mengenai pengelolaan sampah pada anak-anak.

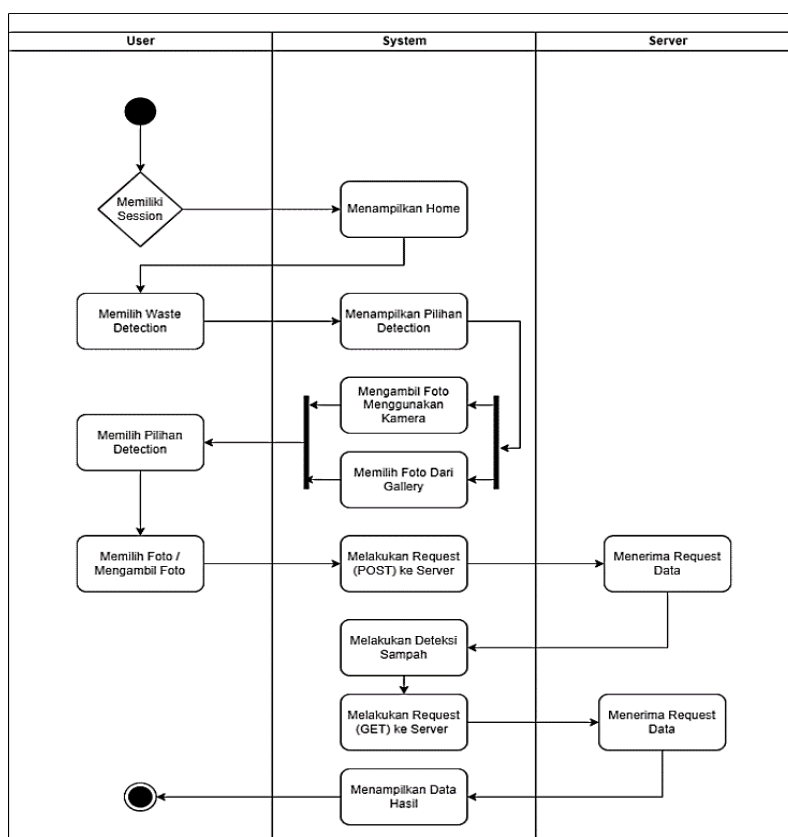
Selanjutnya pengujian aplikasi akan dilakukan dengan dua metode, yaitu *Black Box* dan *Beta Test*. Penggunaan *Black Box* dalam pengujian aplikasi akan memberikan penjelasan tentang kesesuaian harapan dalam pembuatan aplikasi. Sedangkan *Beta Test*, akan dapat mengidentifikasi *bug* atau masalah dalam sistem dan penggunaan aplikasi. Teakhir tahap *operation and maintenance*, dimana aplikasi akan dipelihara dan membenahi berbagai *bugs* yang kemungkinan besar terdapat di dalam aplikasi.

IV. HASIL DAN PEMBAHASAN

Setelah dilakukannya proses penelitian yang melingkupi proses pengembangan dan pengujian, pengembangan aplikasi edukasi pengelolaan sampah pun telah terselesaikan dengan nama aplikasi “*Trashify*”.

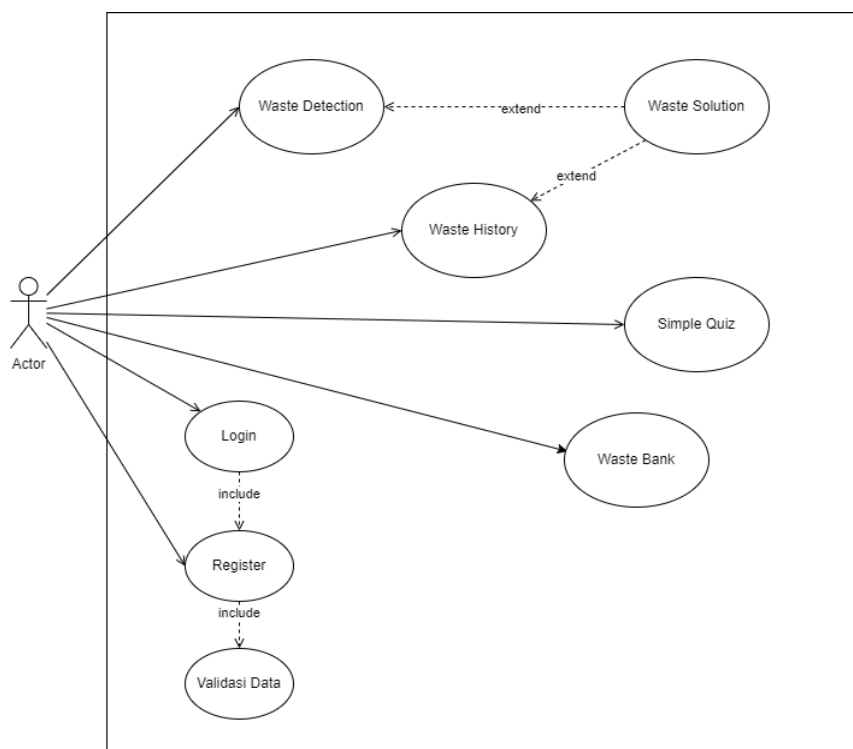
4.1. Rancangan Aplikasi

Berikut merupakan penjelasan mengenai rancangan aplikasi ini. Alur pengguna terlihat seperti pada *activity diagram* di Gambar 2 berikut ini.



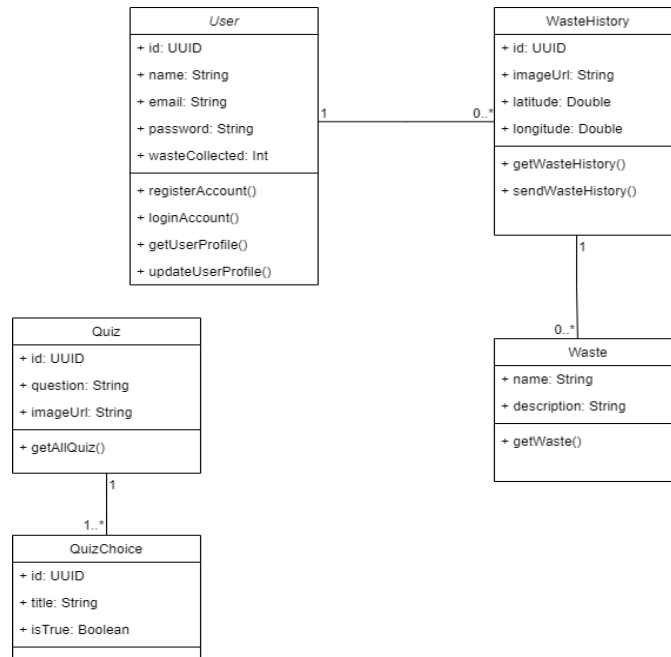
Gambar 2. Activity diagram fitur detection

Berdasarkan *activity diagram* pada Gambar 2, pengguna akan melakukan *login* ke dalam aplikasi dan mendapatkan *session*, yang kemudian akan diarahkan menuju halaman *home*. Pada halaman *home*, pengguna memilih *waste detection* yang selanjutnya akan menampilkan dua pilihan yang perlu dipilih oleh pengguna untuk melakukan deteksi, yaitu melalui pengambilan gambar atau *gallery*. Apabila pengguna memilih untuk mengambil foto baru menggunakan kamera, maka aplikasi akan membuka kamera, sedangkan bila pengguna memilih untuk mengambil foto dari *gallery* maka aplikasi akan membuka *gallery*. Ketika pengguna telah menentukan foto yang dipilih, aplikasi akan mengunggah foto tersebut menuju *server* untuk disimpan dan kemudian dilakukan deteksi. Ketika hasil deteksi diketahui, aplikasi juga akan melakukan *request* pada *server* untuk mendapat info detail berdasarkan hasil deteksi yang didapat yang kemudian akan ditampilkan kepada pengguna.



Gambar 3. *Use case diagram* aplikasi

Selanjutnya Gambar 3 menunjukkan *Use case diagram* merupakan suatu diagram yang bekerja dengan cara mendeskripsikan interaksi antara pengguna suatu sistem dengan sistem itu sendiri melalui sebuah cerita bagaimana sistem tersebut dipakai [14]. *Use case diagram* pada Gambar 3 menjelaskan bahwa aktor yang merupakan pengguna aplikasi dapat mengakses atau menggunakan fitur-fitur yang terdapat dalam aplikasi, mulai dari *login* ke dalam aplikasi hingga melakukan aksi seperti pada fitur *waste detection*.

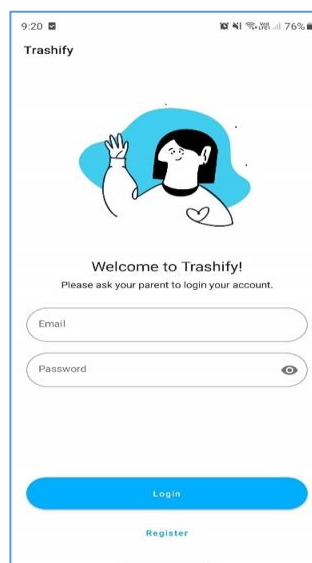


Gambar 4. *Class diagram* aplikasi

Gambar 4 menunjukkan *class diagram* yang merupakan suatu diagram yang digunakan untuk melakukan visualisasi struktur kelas-kelas dari suatu sistem. *Class diagram* juga dapat memperlihatkan hubungan antar kelas dan penjelasan detail setiap kelas di dalam model desain dari suatu sistem [15]. Berdasarkan Gambar 4, setiap objek memiliki hubungan asosiasi dimana hubungan asosiasi merupakan hubungan antar dua kelas yang bersifat statis.

4.2. Hasil Aplikasi

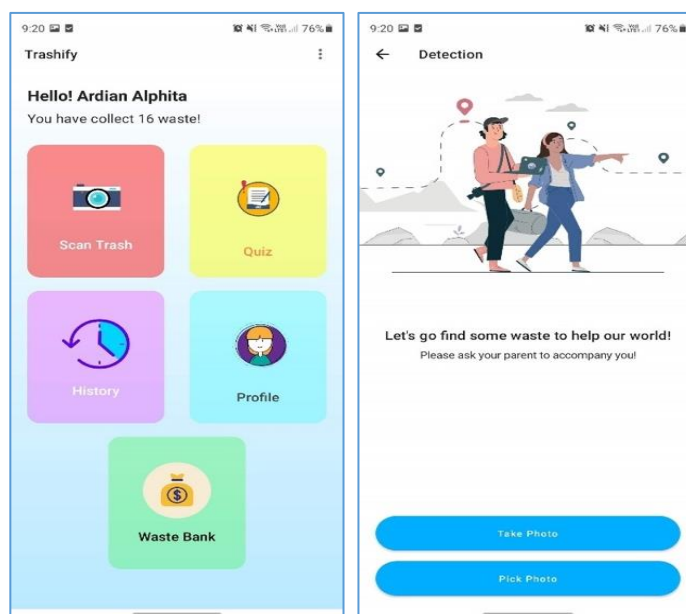
Gambar 5 di bawah ini adalah menunjukkan halaman login aplikasi.



Gambar 5. Tampilan halaman *login*

Untuk dapat menggunakan aplikasi, pengguna perlu membuat akun terlebih dahulu melalui halaman *register*. Akun akan tersimpan di *server* yang kemudian dapat digunakan untuk masuk ke dalam aplikasi. Pengguna hanya perlu melakukan *login* ketika pengguna belum memiliki *session*, ketika telah memiliki *session* maka pengguna tidak perlu login untuk menggunakan aplikasi.

Ketika pengguna berhasil masuk ke dalam aplikasi, maka pengguna akan dapat melihat halaman *home*. Pada halaman *home*, pengguna akan disambut pada bagian *header* yang menampilkan nama dan total sampah yang telah dideteksi oleh pengguna. Pengguna juga dapat melihat dan memilih secara langsung fitur-fitur yang ada pada aplikasi, yaitu deteksi sampah, kuis, *history*, profil pengguna dan bank sampah.

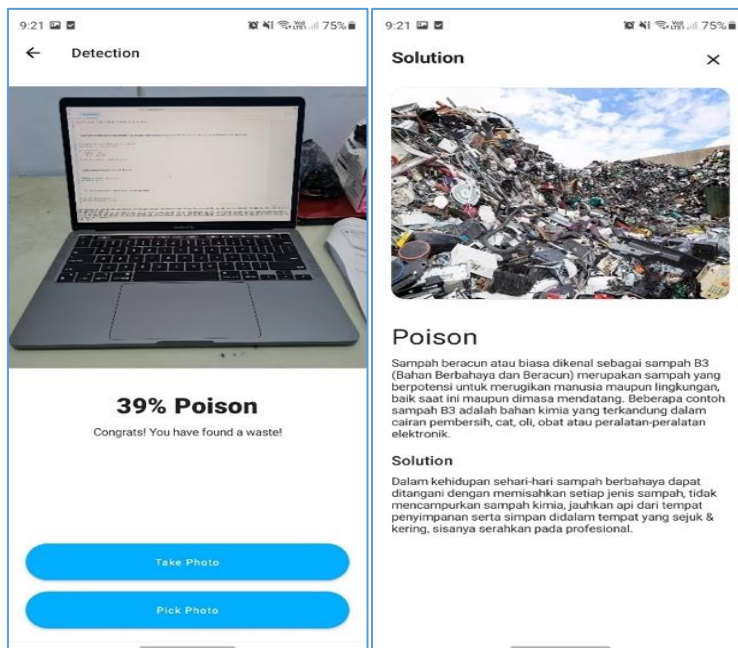


Gambar 6. Tampilan halaman *home* dan *detection*

Selanjutnya pada halaman *detection*, pengguna akan diberikan dua pilihan, yaitu mengambil foto baru menggunakan kamera atau mengambil foto dari *gallery*. Apabila pengguna memilih untuk mengambil foto baru menggunakan kamera, maka aplikasi akan membuka kamera, sedangkan bila pengguna memilih untuk mengambil foto dari *gallery* maka aplikasi akan membuka *gallery*.

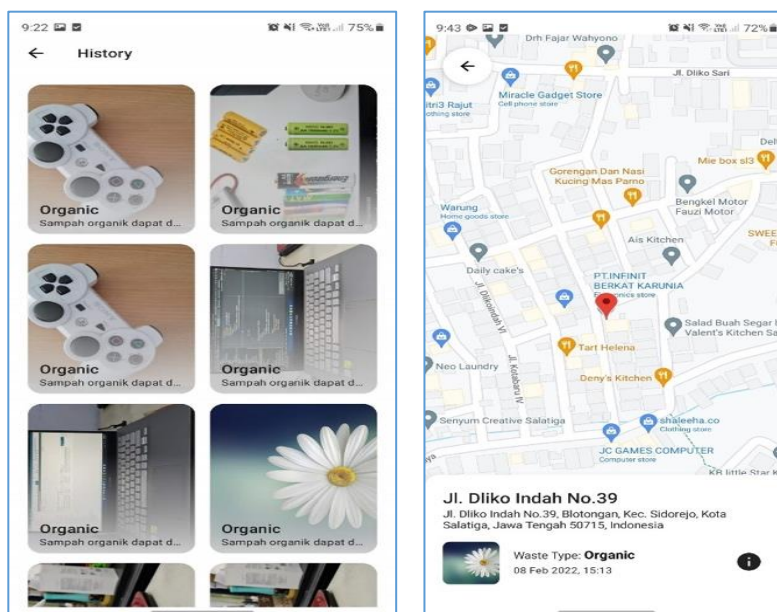
Ketika pengguna telah memilih atau mengambil foto baru, maka aplikasi akan melakukan deteksi sampah pada gambar yang telah dipilih. Ketika aplikasi selesai mendeteksi, maka *bottom sheet* berupa penjelasan mengenai jenis sampah dan cara untuk mengatasi atau mengolah sampah tersebut akan tampil. Apabila *bottom sheet* tersebut ditutup maka pengguna akan kembali ke halaman *detection*, namun dengan persentase kemungkinan jenis sampah yang telah berhasil dideteksi sebelumnya. Tampilan bisa dilihat pada Gambar 7.

Selanjutnya halaman *history* merupakan halaman yang berisikan seluruh foto yang telah diambil oleh pengguna. Pengguna dapat melihat foto apa saja yang telah diambil dan dideteksi, jenis sampah yang terdeteksi dan penjelasan singkat mengenai jenis sampah tersebut. Bila pengguna memilih salah satu foto yang ada pada daftar *history* tersebut, pengguna akan diarahkan menuju halaman *history detail*



Gambar 7. Tampilan halaman deteksi sampah

Gambar 8 menunjukkan halaman *history detail* yang memperlihatkan detail dari sampah yang telah terdeteksi. Pengguna dapat melihat dimana foto sampah tersebut diambil melalui *Google Map* yang ditampilkan beserta dengan alamat lengkapnya, waktu pengambilan foto dan jenis sampah yang terdeteksi. Pengguna juga dapat melihat kembali mengenai penjelasan jenis sampah beserta cara pengolahan sampah tersebut melalui tombol info pada bagian bawah.



Gambar 8. Tampilan halaman *history* dan *history detail*

4.3. Kode Program Pembuatan Model

Kode Program 1 merupakan kode yang digunakan untuk pembuatan model *machine learning* menggunakan bahasa pemrograman Python. Dalam pembuatan model ini,

digunakan *library* Tensorflow, sehingga dalam pembuatan model ini nantinya akan menghasilkan *file* dengan format *.tflite*. Pembuatan model ini dilakukan dengan menggunakan metode Transfer Learning, yaitu merupakan suatu teknik yang memanfaatkan model yang sudah ada dan telah dilatih sebelumnya (*pretrained*) yang kemudian digunakan untuk mengklasifikasikan dataset baru sehingga akan mempersingkat proses yang ada karena tidak perlu melakukan *training* dari awal [16].

```
!unzip -q "/content/drive/MyDrive/Tugas Akhir/TRAIN.zip"

data_path = pathlib.Path("/content/TRAIN")
data = DataLoader.from_folder(data_path)
train_data, test_data = data.split(0.9)

model = image_classifier.create(train_data)
loss, accuracy = model.evaluate(test_data)

model.export(export_dir='/mm_waste')
```

Kode 1. *Coding* untuk pembuatan model

Proses pembuatan model tersebut diawali dengan melakukan membuka *zip file* yang berisikan dataset foto yang telah disiapkan. Kemudian, aspek terpenting dalam melatih suatu model dalam *machine learning* adalah dengan tidak menggunakan semua data yang ada untuk dilatih. Sehingga perlu memisahkan data yang akan dilatih dan digunakan untuk *test*. Dalam kode program 1, dengan menggunakan sintaks “*data.split(0.9)*”, akan mendapatkan 90% data yang akan dilatih sedangkan sisanya digunakan untuk *test*. Setelah memisahkan data, dapat dilakukan *training* dan *test* untuk mendapatkan perkiraan akurasi yang didapatkan melalui *dataset* yang telah disediakan.

```
Model: "sequential"
-----
Layer (type)                Output Shape              Param #
-----
hub_keras_layer_v1v2 (HubKe (None, 1280)              3413024
rusLayerV1V2)
dropout (Dropout)          (None, 1280)              0
dense (Dense)               (None, 4)                 5124
-----
Total params: 3,418,148
Trainable params: 5,124
Non-trainable params: 3,413,024
-----
None
/usr/local/lib/python3.7/dist-packages/keras/optimizer_v2/gradient_descent.py:102: UserWarning: The "lr" argument is deprecated, use "learning_rate" instead.
  super(SG, self).__init__(name, **kwargs)
Epoch 1/5
638/638 [=====] - 688s 1s/step - loss: 0.6389 - accuracy: 0.8612
Epoch 2/5
638/638 [=====] - 661s 1s/step - loss: 0.5734 - accuracy: 0.8953
Epoch 3/5
638/638 [=====] - 663s 1s/step - loss: 0.5671 - accuracy: 0.8990
Epoch 4/5
638/638 [=====] - 666s 1s/step - loss: 0.5603 - accuracy: 0.9025
Epoch 5/5
638/638 [=====] - 662s 1s/step - loss: 0.5559 - accuracy: 0.9050

71/71 [=====] - 84s 991ms/step - loss: 0.5468 - accuracy: 0.9132
```

Gambar 9. Hasil *training* dan *test dataset*

Melalui Gambar 9, terlihat bahwa hasil *training* mendapat akurasi sebesar 90%. Sedangkan hasil *test* yang digunakan untuk mengukur perkiraan ketika digunakan dalam aplikasi nantinya adalah mendapatkan akurasi sebesar 91%. Ketika telah mendapatkan hasil prediksi yang cukup tinggi, langkah terakhir yang dilakukan adalah melakukan eksport *file model* sehingga dapat digunakan dalam aplikasi.

Selanjutnya Kode Program 2 merupakan fungsi yang digunakan untuk melakukan deteksi pada foto yang telah diambil atau dipilih sebelumnya yang ditulis menggunakan bahasa pemrograman Kotlin. Deteksi dilakukan secara *on-device* untuk lebih mempercepat proses deteksi, sehingga diperlukan model yang telah dibuat sebelumnya

```
fun detectImage(
    localModel: LocalModel,
    uriImg: Uri,
    context: Context,
) {
    _detection.value = Resource.Loading()

    val options = CustomImageLabelerOptions.Builder(localModel)
        .setConfidenceThreshold(0.0f)
        .setMaxResultCount(5)
        .build()

    val labeler = ImageLabeling.getClient(options)
    val image = InputImage.fromFilePath(context, uriImg)

    viewModelScope.launch {
        delay(2000)
        labeler.process(image)
            .addOnSuccessListener { labels ->
                val label = labels.first()
                val text = label.text
                val confidence = label.confidence

                result.value = "$text $confidence"
                _detection.value = Resource.Success(true)
            }
            .addOnFailureListener { exception ->
                result.value = exception.message.orEmpty()
                _detection.value = Resource.Error(exception)
            }
            .addOnCanceledListener {
                result.value = "cancel"
                _detection.value = Resource.Success(true)
            }
    }
}
```

Kode 2. Kode program untuk melakukan deteksi pada aplikasi

Kode Program 3 merupakan kode yang digunakan untuk mendapatkan model yang telah dilatih sebelumnya dan dibangun menjadi *LocalModel* agar dapat digunakan sebagai input *value* dalam parameter fungsi *detectImage()*. Setelah mendapatkan *LocalModel*, langkah selanjutnya adalah dengan mengatur objek “*option*” sebelum dilakukan proses deteksi melalui kode program berikut.

```
val localModel = LocalModel.Builder()
    .setAssetFilePath("model_2.tflite")
    .build()
```

Kode 3: Kode program untuk menggunakan model hasil *training*

Pada objek “*option*” tersebut, dilakukan konfigurasi untuk mengatur batas *confidence* atau akurasi yang akan diterima, pada aplikasi ini tidak diberikan batasan sehingga hasil akurasi yang akan didapatkan berada dalam skala 0 sampai 100 persen. Kemudian dapat dilakukan proses deteksi secara asynchronous menggunakan *viewModelScope*. Proses deteksi ini akan mengembalikan objek *ImageLabel* yang berisikan nama label yang telah dilatih sebelumnya dan juga hasil *confidence* yang didapatkan berdasarkan foto yang diambil.

```

val options = CustomImageLabelerOptions.Builder(localModel)
    .setConfidenceThreshold(0.0f)
    .setMaxResultCount(5)
    .build()
val labeler = ImageLabeling.getClient(options)
val image = InputImage.fromFilePath(context, uriImg)

```

Kode 4: Kode program untuk mengatur objek *option*

4.4. Pengujian

Pengujian pada aplikasi ini dilakukan dengan menggunakan metode *Black Box* dan juga *Beta Test*. Untuk pengujian menggunakan metode *Black Box*, berikut hasil pengujian yang didapatkan:

Tabel 1 Data pengujian *black box*

Skenario	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
Membuat akun baru dan melakukan <i>login</i>	Pengguna dapat masuk ke dalam aplikasi menggunakan akun yang dibuat	Pengguna berhasil masuk ke dalam aplikasi menggunakan akun yang dibuat	Valid
Melakukan deteksi sampah elektronik berupa laptop	Mendapatkan hasil berupa sampah beracun	Hasil yang didapat adalah sampah beracun	Valid
Melakukan deteksi sampah organik berupa dedaunan	Mendapatkan hasil berupa sampah organik	Hasil yang didapat adalah sampah organik	Valid
Melakukan deteksi sampah non-organik berupa botol	Mendapatkan hasil berupa sampah non-organik	Hasil yang didapat adalah sampah non-organik	Valid
Mengerjakan soal-soal kuis	Pengguna dapat menyelesaikan kuis dan kembali ke halaman <i>Home</i>	Pengguna berhasil menyelesaikan kuis dan kembali ke halaman <i>Home</i>	Valid
Membuka <i>waste history</i> beserta dengan detailnya	Pengguna dapat melihat seluruh sampah yang telah dideteksi dan dapat mengetahui lokasi beserta jenis sampah yang terdeteksi	Pengguna berhasil melihat seluruh sampah yang telah dideteksi dan dapat mengetahui lokasi beserta jenis sampah yang terdeteksi	Valid
Membuka <i>waste bank</i>	Pengguna dapat melihat daftar bank sampah yang berada di dekatnya dan dapat membuka <i>maps</i> untuk melakukan navigasi	Pengguna berhasil melihat daftar bank sampah yang berada di dekatnya dan dapat membuka <i>maps</i> untuk melakukan navigasi	Valid

Pengujian yang kedua, digunakan metode *Beta Test* dengan mengirim aplikasi kepada beberapa penguji dan memberikan Google Form untuk mengetahui pengalaman

pengguna dan menerima masukan dari penguji. *Beta Test* dilakukan secara terbatas sehingga pengujian yang dilakukan merupakan *closed beta test* dimana pengujian hanya dilakukan pada sejumlah pengguna yang telah dipilih. Pengujian dilakukan sebanyak 10 orang dari beberapa kalangan, yaitu mahasiswa TI, mahasiswa non-TI, dan juga orang tua. Hal ini dilakukan agar dapat mendapatkan kritik dan saran untuk mengembangkan aplikasi jadi lebih baik lagi.

Sebelum penguji melakukan pengujian pada aplikasi, penguji dijelaskan secara singkat mengenai gambaran besar aplikasi yang telah dikembangkan. Penguji menerima penjelasan bahwa aplikasi yang dikembangkan merupakan aplikasi edukasi mengenai pengelolaan sampah yang ditujukan untuk anak di kalangan sekolah dasar. Hal ini bertujuan agar penguji paham mengenai target pengguna yang diharapkan sehingga dapat memberikan kritik dan saran yang lebih spesifik kepada target pengguna tersebut.

Setiap penguji nantinya akan mendapatkan tiga buah pertanyaan, yaitu 1) Bagaimana pengalaman anda dalam menggunakan aplikasi Trashify? 2) Apakah anda menemukan *error* atau kesalahan sistem ketika menggunakan aplikasi Trashify? Jika iya, boleh dijelaskan! 3) Hal atau fitur apa yang dapat ditingkatkan dari aplikasi Trashify? Ketiga pertanyaan ini akan dijawab dan dicatat peneliti untuk selanjutnya diolah kembali.

Data yang dikumpulkan dari hasil pengujian merupakan data kualitatif, sehingga sebelum data yang didapat dapat diambil suatu kesimpulan perlu dilakukan suatu proses yang dinamakan reduksi data. Reduksi data merupakan proses pemilahan, pemusatan perhatian pada penyederhanaan, pengabstrakan dan transformasi data kasar yang muncul dari catatan-catatan tertulis dalam lapangan. Dalam proses reduksi data sendiri meliputi beberapa proses, yaitu: (1) meringkas data, (2) mengkode, (3) menelusuri tema, dan (4) membuat gugus-gugus. Proses tersebut dapat dilakukan dengan cara melakukan seleksi pada data, membuat ringkasan atau uraian singkat, dan menggolongkannya ke dalam pola yang lebih luas [15].

Setelah dilakukan reduksi data, didapatkan beberapa kategori atau poin penting, yaitu (1) Objek deteksi, (2) Konfirmasi, (3) Tampilan aplikasi, (4) Fitur tambahan, dan (5) Permasalahan minor. Melalui beberapa kategori tersebut, maka kritik atau masalah yang dialami oleh penguji dapat dijelaskan lebih detail, sebagai berikut:

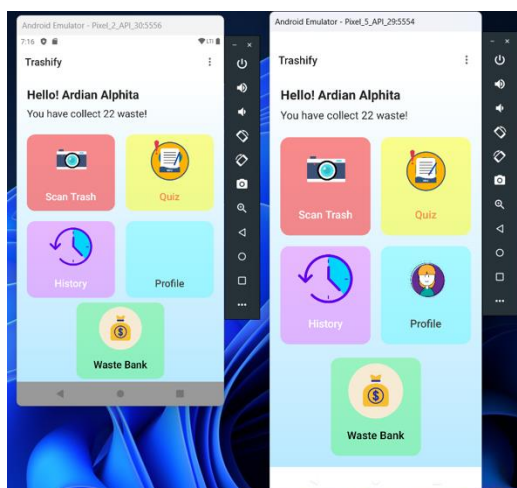
1. Dalam proses registrasi akun, tidak memunculkan pemberitahuan bahwa registrasi telah berhasil sehingga membuat pengguna tidak tahu langkah apa yang perlu dilakukan setelah melakukan registrasi serta pada proses *logout* tidak diberikan konfirmasi.
2. Proses deteksi yang belum akurat, sehingga tidak mendapatkan hasil sesuai dengan ekspektasi pengguna dikarenakan data model yang kurang lengkap.
3. Tampilan aplikasi belum dapat beradaptasi dengan ukuran perangkat yang berbeda, sehingga mengganggu pengalaman pemakaian pengguna.
4. Dan beberapa masalah *minor* yang kemungkinan besar terjadi karena masalah perangkat atau koneksi, seperti proses pengunggahan gambar deteksi memakan waktu lama, sehingga pengguna tidak dapat melihat barang yang telah dideteksi pada halaman *history* dan tampilan *maps* yang tidak dapat menampilkan peta dan alamat.

Kemudian, penguji juga memberikan saran yang diharapkan dapat membuat aplikasi menjadi baik lagi, antara lain:

1. Penambahan objek deteksi, agar hasil deteksi menjadi lebih akurat.
2. Pemberian konfirmasi dalam setiap proses, seperti pada saat melakukan registrasi dan *logout*.
3. Penambahan waktu kuis di setiap soal, dengan pertimbangan target pengguna merupakan anak sekolah dasar.
4. Penambahan *mini-game* dan fitur sosial pertemanan untuk membuat aplikasi menjadi lebih interaktif.
5. Peningkatan *User Interface* (UI) agar dapat beradaptasi dengan berbagai ukuran perangkat.

Setelah mendapatkan kritik dan saran dari penguji, dilakukan perbaikan minor dengan tujuan untuk meningkatkan kualitas aplikasi versi 1.0. Untuk saran mengenai penambahan fitur berupa permainan atau sosial, dapat dikembangkan pada proses pengembangan untuk *major update* berikutnya. Berikut merupakan beberapa perbaikan yang telah dilakukan untuk aplikasi versi 1.0, yaitu:

1. Memperbaiki alur aplikasi sehingga pengguna tidak merasa kebingungan pada navigasi di dalam aplikasi, dengan cara menambahkan pesan pada beberapa fitur seperti registrasi dan *logout*.
2. Memperbaiki UI agar lebih mudah beradaptasi pada berbagai ukuran layar perangkat, seperti pada halaman *home*, dimana sebelumnya terdapat tombol yang terpotong pada layar perangkat yang kecil, sekarang dapat beradaptasi.



Gambar 10. Penyesuaian halaman home pada perangkat yang lebih kecil

3. Menambahkan data sampel pada *dataset* untuk meningkatkan akurasi pada model. Setelah dilakukan penambahan data sampel, akurasi model yang digunakan menjadi 90%, meski menurun dari model yang sebelumnya, namun dengan model yang baru, pengguna dapat mendeteksi beberapa barang yang sebelumnya tidak dapat dideteksi seperti masker.

```
71/71 [=====] - 83s 1s/step - loss: 0.5579 - accuracy: 0.9080
```

Gambar 16. Hasil test *dataset* setelah penambahan data

Selanjutnya adalah penyesuaian *user experience* pada fitur kuis agar lebih mudah digunakan oleh anak sekolah dasar dengan menambahkan waktu pada *timer*.

V. SIMPULAN DAN SARAN

Berdasarkan penelitian yang telah dilakukan, serta berdasarkan hasil pengujian yang didapat, dapat disimpulkan bahwa Pengembangan Aplikasi Mobile tentang Edukasi Pengelolaan Sampah untuk Anak Sekolah Dasar dapat dijadikan sebagai sarana untuk membantu anak-anak dalam memahami betapa pentingnya pengelolaan sampah. Meskipun di dalam aplikasi masih memiliki fitur yang terbatas, namun aplikasi ini dinilai tetap dapat membantu anak-anak agar dapat mudah memahami edukasi mengenai pengelolaan sampah dengan cara yang interaktif. Kemudian, beberapa hal yang dapat ditingkatkan pada aplikasi ini adalah melakukan penambahan data berupa foto dalam *dataset* agar hasil deteksi lebih beragam dan lebih akurat. Kemudian, dapat meningkatkan pengalaman pengguna atau *user experience* dalam proses deteksi sampah dengan dilakukan secara *real-time* di dalam kamera, sehingga pengguna tidak perlu menekan beberapa tombol terlebih dahulu untuk dapat mengetahui hasil deteksi. Aplikasi ini pun juga masih memiliki peluang untuk dikembangkan lebih jauh lagi dengan menambah fitur-fitur yang relevan agar dapat semakin membantu anak-anak dalam belajar secara interaktif, seperti penambahan permainan agar aplikasi menjadi lebih interaktif serta penambahan fitur sosial agar anak-anak dapat berkolaborasi dan bermain bersama dengan teman-temannya.

UCAPAN TERIMA KASIH

Ucapan terima kasih kepada Universitas Kristen Satya Wacana yang telah membantu dalam proses penelitian ini. Ucapan terima kasih juga dilayangkan kepada pembimbing, teman-teman serta keluarga yang selalu mendukung selama penelitian ini berlangsung.

DAFTAR PUSTAKA

- [1] Kementerian Lingkungan Hidup dan Kehutanan, "SIPSN - Sistem Informasi Pengelolaan Sampah Nasional." [Daring]. Available: <https://sipsn.menlhk.go.id/sipsn/>
- [2] R. Rondiyah, S. Sulistiyani, dan M. Rahardjo, "Faktor-Faktor Yang Mempengaruhi Kinerja Pengelolaan Sampah Di Pasar Banjarsari Kota Pekalongan," *Jurnal Kesehatan Masyarakat Universitas Diponegoro*, vol. 2, no. 3, hlm. 192–199, 2014.
- [3] Asosiasi Penyelenggara Jasa Internet Indonesia, "Survei Profil Internet Indonesia 2022." [Daring]. Available: <https://apjii.or.id/survei>
- [4] Ade. Bastian, Yofi. Awwaluddin, Whydiantoro., dan Budiman., "Designing Environment Care Adventure Game Based on Android Using Construct 2," *Jurnal Mantik*, vol. 4, no. 3, hlm. 1–7, 2020, [Daring]. Available: <http://iocscience.org/ejournal/index.php/mantik/article/view/882/595>

- [5] D. Fransisca dan R. N. Yusuf, "Jurnal Kesehatan Medika Saintika," *Jurnal Kesehatan Medika Saintika Volume*, vol. 10, no. 2, hlm. 11–24, 2018.
- [6] F. Alfian, "Rancang Bangun Robot Pemilah Sampah Organik Dan Non Organik," *Skripsi Teknik Informatika*, 2019.
- [7] M. Mustika, E. P. A. Sugara, dan M. Pratiwi, "Pengembangan Media Pembelajaran Interaktif dengan Menggunakan Metode Multimedia Development Life Cycle," *Jurnal Online Informatika*, vol. 2, no. 2, hlm. 121, 2018, doi: 10.15575/join.v2i2.139.
- [8] Mauji. Daulana dan Hanhan. Maulana, "The Development of Go Go Greenland Educational Environmental Game Based on Android," no. 112, hlm. 2–9, 2017, [Daring]. Available: <https://elib.unikom.ac.id/files/disk1/756/jbptunikompp-gdl-maujidaula-37793-11-21.unik-s.pdf>
- [9] H. Rahmayanti, V. Oktaviani, dan Y. Syani, "Development of sorting waste game android based for early childhood in environmental education," *J Phys Conf Ser*, vol. 1434, no. 1, 2020, doi: 10.1088/1742-6596/1434/1/012029.
- [10] A. Roihan, P. A. Sunarya, dan A. S. Rafika, "Pemanfaatan Machine Learning dalam Berbagai Bidang: Review paper," *IJCIT (Indonesian Journal on Computer and Information Technology)*, vol. 5, no. 1, hlm. 75–82, 2020, doi: 10.31294/ijcit.v5i1.7951.
- [11] A. L. LESTARI, "Sistem Informasi Pemesanan dan Layanan Antar Makanan Sesurabaya Berbasis Android," 2017.
- [12] H. A. Zainal, "Metodologi Penelitian Pada Bidang Teknologi Informasi," *Metodologi Penelitian Pada Bidang Ilmu Komputer Dan Teknologi Informasi*, vol. 4(1), hlm. 126–130, 2007.
- [13] S. Maesaroh, D. Rohmayani, Ramlan, dan Arsul, "Rancang Bangun Sistem Informasi Kepegawaian (SIMPEG) dengan SDLC Metode Waterfall Studi Kasus di Kantor BKPLD Kabupaten Tasikmalaya," *Technical Education Development Center (TEDC)*, vol. 11, no. 2, hlm. 197–202, 2017.
- [14] T. B. Kurniawan Syarifuddin, "Perancangan Sistem Aplikasi Pemesanan Makanan dan Minuman Pada Cafeteria NO Caffe di TANjung Balai Karimun Menggunakan Bahasa Pemrograman PHP dan My.SQL," *J Chem Inf Model*, vol. 53, no. 9, hlm. 1689–1699, 2020.
- [15] A. Rijali, "Analisis Data Kualitatif," *Alhadharah: Jurnal Ilmu Dakwah*, vol. 17, no. 33, hlm. 81, 2019, doi: 10.18592/alhadharah.v17i33.2374.
- [16] F. Rochman dan H. Junaedi, "Implementasi Transfer Learning untuk Identifikasi Ordo Tumbuhan melalui Daun," *Jurnal Syntax Admiration*, vol. 1, no. 6, hlm. 672–679, 2020.