

Implementasi *Text Recogniter* yang Diterjemahkan ke Bahasa Lain dengan *Firestore ML-Kit* Berbasis Android

Pascal Aditia Muclis¹, Somantri²

^{1,2} Universitas Nusa Putra, Jln Raya Cibatu, Cisaat No.21, Sukabumi, Jawa Barat 43155

e-mail: ¹pascal.aditia_TI18@nusaputra.ac.id, ²somantri@nusaputra.ac.id

Submitted Date: July 17th, 2021
Revised Date: August 09th, 2021

Reviewed Date: July 25th, 2021
Accepted Date: August 11th, 2021

Abstract

Each community area has information, announcements and news made in a picture, pamphlet or poster. However, there are a lot of foreign languages in the information that are not understood by most people because of the lack of literacy in foreign language learning for the community, for that the author makes research on text recognition. Recognition or detection of text characters is also known as optical character recognition (OCR), which converts handwritten or printed text into data that can be edited and read by the system. After the text is converted into written form, the text is then converted from a foreign language (English) and then converted become Indonesian. This study uses observation and literature study in collecting information and materials for research, while for the method of making the system using prototyping which the system can be developed more quickly. The results of this application can help the people of Gunungendut Village to overcome the problem of understanding foreign language info that is in the image so that it is easy to understand using a mobile application.

Keywords: Android; Text Recogniter; Translate

Abstrak

Setiap daerah masyarakat memiliki info, pengumuman dan berita yang dibuat didalam sebuah gambar, pamflet atau poster. Namun banyak sekali terdapat bahasa asing di dalam info tersebut yang tidak dipahami oleh kebanyakan masyarakat karena kurangnya literasi pembelajaran bahasa asing bagi masyarakat, untuk itu penulis membuat penelitian tentang pengenalan teks. Pengenalan atau pendeteksian karakter teks di kenal juga dengan *optical character recognition (OCR)* yaitu mengubah tulisan tangan atau teks cetak menjadi data yang dapat diedit dan dibaca oleh sistem, Setelah teks diubah ke dalam bentuk tulisan kemudian teks tersebut diubah dari bahasa asing (inggris) kemudian diubah menjadi bahasa indonesia. Penelitian ini menggunakan observasi dan studi pustaka dalam mengumpulkan informasi dan bahan untuk peneltian, sedangkan untuk metode pembuatan sistem menggunakan *prototyping* yang di mana sistem bisa lebih cepat dikembangkan. Hasil aplikasi ini dapat membantu masyarakat Desa Gunungendut untuk mengatasi masalah memahami info bahasa asing yang berada di dalam gambar agar mudah dipahami dengan menggunakan aplikasi *mobile*.

Keywords: Android; Text Recogniter; Terjemahan

1 Pendahuluan

Kecanggihan teknologi informasi semakin berkembang pesat dan dapat dimanfaatkan untuk mempermudah dan menghemat biaya. Salah satu contohnya yaitu penggunaan perangkat lunak aplikasi, perangkat lunak atau software adalah sistem yang berada di dalam sebuah mesin, salah satunya adalah program aplikasi. Perangkat lunak

disebut juga sebagai bagian sistem yang tidak berwujud, beberapa perangkat lunak dapat diakses pada platform Android yang merupakan generasi *mobile phone* tercanggih dan sangat banyak diminati oleh penduduk negara Indonesia bahkan di dunia sehingga menjadikannya salah satu teknologi tercanggih di dunia dan sebagai generasi baru *mobile*

phone yang membuat tertarik dunia untuk menggunakan dan mengembangkannya.

Perusahaan terbesar yang bernama Google memiliki produk yang ditawarkan bernama *firebase*, di dalam *firebase* tersebut terdapat modul yang bernama *ML-Kit (Machine Learning Kit)*. *ML Kit* adalah *mobile SDK* yang memberikan pembelajaran mesin Google (*Google's Machine Learning*) untuk sistem aplikasi Android dan iOS dalam paket yang *powerfull*, fitur ini mudah digunakan untuk dikembangkan dalam aplikasi android dengan *Machine Learning*, di dalam *ML-Kit* terdapat beberapa fitur di antaranya *Text Recogniter* yang digunakan untuk mengubah teks cetak ke dalam sistem android dan *Translate* yang berfungsi untuk menerjemahkan bahasa dari teks (Ernawati, Wati, & Maulana, 2021).

Pengenalan atau pendektesian karakter teks dikenal juga dengan *optical character recognition (OCR)* yaitu mendeteksi teks yang berada di dalam kertas ataupun teks cetak yang kemudian akan diubah menjadi teks yang dapat diedit pada sistem aplikasi, *Optical Character Recognition (OCR)* merupakan sistem komputer yang dapat digunakan untuk mendeteksi teks atau angka untuk diubah ke dalam bentuk file teks.

Penelitian yang terkait pengembangan sistem *text recogniter* berbasis android telah dilakukan beberapa peneliti. Penelitian oleh M. Arinal Haqil Ghifari, Andi Susilo (Ghifari & Susilo, 2020) merancang sistem pendektesian atau pengenalan teks dengan menggunakan metode pengembangan *System Development Life Cycle*. Hasil dari penelitian tersebut membuat sistem pengenalan teks yang diubah menjadi audio untuk membantu lansia dalam memahami teks.

Penelitian Firmanda Mulyana Nugroho, Agi Putra Kharisma dan Wibisono Sukmo Wardhono (Nugroho, Kharisma, & Wardhono, 2019) merancang sistem pengembangan aplikasi pembelajaran kanji. Sistem ini menggunakan *firebase ML-KIT* untuk *text recogniternya* yang kemudian *text* tersebut di ubah kedalam suara (*text to speech*) dari *Kanji Alive API*. Hasil dari penelitian tersebut memberikan efektifitas pembelajaran bahasa jepang atau kanji.

Penelitian Toby Zulkarnain, Suharyanto dan Anton (Zulkarnain, Suharyanto, & Anton, 2020) merancang sistem serupa tentang *optical character recogniter* berbasis android. Sistem tersebut menggunakan fitur *firebase ML-KIT* dengan algoritma *backpropagation*, metode yang digunakan untuk mengembangkan sistem adalah metode *waterfall*. Hasil dari penelitian tersebut

sistem dapat mendeteksi teks dari sebuah gambar yang kemudian di ubah kedalam teks digital.

Setiap daerah masyarakat memiliki info, pengumuman dan berita yang dibuat didalam sebuah gambar, pamflet atau poster. Namun banyak sekali terdapat bahasa asing di dalam info tersebut yang tidak dipahami oleh kebanyakan masyarakat, karena kurangnya literasi pembelajaran bahasa asing bagi masyarakat, Untuk itu penulis membuat penelitian "Implementasi *Text Recogniter* Yang Diterjemahkan Ke Bahasa Lain Dengan *Firebase ML-Kit* Berbasis Android". Tujuan dari penulis membuat penelitian ini adalah agar menghasilkan sistem aplikasi yang dapat membantu masyarakat Desa Gunungendut dalam mengatasi permasalahan memahami info bahasa asing yang berada di dalam gambar agar mudah dipahami dengan menggunakan aplikasi android.

2 Metode Penelitian

Teknik pengumpulan penelitian ini di antaranya adalah observasi dan studi pustaka. Pada tahap Observasi peneliti mengumpulkan informasi dengan memperhatikan kegiatan seseorang dalam menerjemahkan teks di dalam sebuah gambar. Studi pustaka dilakukan untuk mencari pemahaman informasi tentang *text recogniter* dengan mencari beberapa jurnal yang berkaitan dengan pembahasan yang diperlukan untuk dijadikan referensi dalam penelitian ini.

Adapun metode penelitian untuk pengembangan sistem ini adalah prototyping, Prototyping adalah metode *lifecycle* sistem yang berdasarkan konsep model bekerja (*working model*). Tujuannya adalah membuat metode atau model menjadi sistem final. Yang berarti sistem dapat dikembangkan dengan lebih cepat dari pada metode lainnya (Fikriyya & Dirgahayu, 2020). Tahapan dalam pembuatan prototyping adalah:

- Analisis kebutuhan, pada tahap ini *client* dan pengembang saling menganalisis kebutuhan software, menganalisis semua kebutuhan sistem yang akan dibuat.
- Perancangan prototyping, pada tahap ini pengembang membuat rancangan sistem seperti membuat *UML*.
- Evaluasi prototyping, pada tahap ini sistem yang sudah dirancang akan dievaluasi kembali apakah sudah sesuai dengan yang diharapkan, jika sudah tahap ini akan dilewati.
- Pengkodean sistem, pada tahap ini sistem yang sudah selesai dirancang akan diterjemahkan kedalam bahasa pemrograman seperti *kotlin*, *java* dan *xml*.

- e. Pengujian sistem, pengujian sistem akan dilakukan dengan menggunakan *black box* untuk mengetahui hasil sistem.
- f. Evaluasi sistem, pada tahap ini sistem akan dievaluasi kembali apakah masih terdapat kesalahan atau error, jika sistem sudah sesuai maka akan dilanjutkan ke tahap terakhir.
- g. Penggunaan sistem, pada tahap ini perangkat lunak atau sistem yang sudah diuji dan diterima *client* siap dipakai.

3 Hasil dan Pembahasan

3.1 Analisis Kebutuhan

Aplikasi mencakup:

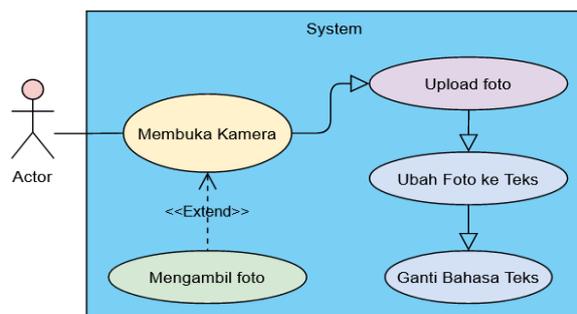
- a. Halaman *Splash Screen*
- b. Halaman *dashboard*
- c. Fitur mengambil gambar dari kamera
- d. Menampilkan gambar yang sudah diambil dari kamera
- e. Tombol mengubah gambar ke tulisan
- f. Halaman menampilkan hasil tulisan
- g. Tombol pengubah bahasa

3.2 Membangun Prototyping

Membangun prototyping dimulai dengan merancang sistem menggunakan UML, UML merupakan singkatan dari *Unified Modelling Language* yaitu metode yang menggambarkan rancangan sistem atau model (Suendri, 2018). Definisi UML adalah sebagai suatu rancangan yang sudah menjadi standar visualisasi, Berikut ini UML dari sistem peneliti:

a. Use Case Diagram

Use case diagram merupakan diagram yang mendefinisikan rancangan *relation* antara *user* dengan sistem. *Use case* aplikasi ini dimulai dengan aktor yang membuka kamera, memilih foto, kemudian mengunggah (*upload*) foto tersebut, foto akan di ubah kedalam tulisan dan mengubah bahasa teks tersebut. Adapun *use case system* terdapat di **Gambar 1**.



Gambar 1. Use Case Diagram

b. Use Case Skenario

Use case skenario membuka kamera terdapat di **Tabel 1**.

Tabel 1. Skrenario Membuka Kamera

Tujuan	Membuka kamera dan mengambil foto
Aktor	User
Kondisi awal	Aksi tidak ada
Skenario Utama	User sukses membuka kamera
Skenario Alternatif	1. Kamera gagal terbuka karena kamera rusak 2. Kamera gagal terbuka karena sistem error
Kondisi Akhir	Kamera bisa terbuka dan bisa mengambil foto

Use Case skenario Upload Foto terdapat di **Tabel 2**.

Tabel 2. Skenario Upload Foto

Tujuan	Mengunggah (<i>upload</i>) foto yang diambil dari kamera
Aktor	User
Kondisi awal	Aksi tidak ada
Skenario Utama	User berhasil mengunggah (<i>upload</i>) foto
Skenario Alternatif	1. Upload foto gagal karena jaringan error 2. Uploaf foto gagal karena sistem error
Kondisi Akhir	Foto akan tampil di halaman dashboard

Use Case Skenario Mengubah foto ke teks terdapat pada **Tabel 3**.

Tabel 3. Skenario Mengubah foto ke teks

Tujuan	Mengubah foto menjadi tulisan
Aktor	User
Kondisi awal	Foto tidak berubah
Skenario Utama	Foto berhasil berubah menjadi teks
Skenario Alternatif	1. Foto tidak berubah menjadi teks karena jaringan error 2. Foto tidak berubah menjadi teks karena sistem error
Kondisi Akhir	Hasil teks akan ditampilkan

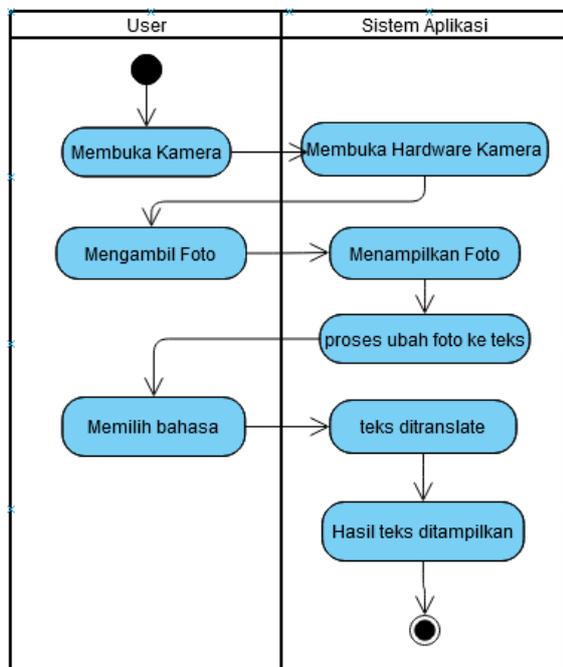
Use Case Skenario ganti bahasa teks terdapat di **Tabel 4**.

Tabel 4. Skenario ganti bahasa teks

Tujuan	Mengganti bahasa teks
Aktor	User
Kondisi awal	Teks tidak berubah
Skenario Utama	Bahasa teks berhasil diubah
Skenario Alternatif	1. Bahasa teks tidak berubah karena jaringan error 2. Bahasa teks tidak berubah karena sistem error
Kondisi Akhir	Teks berhasil ditampilkan dengan bahasa yang di <i>translate</i>

c. Activity Diagram

Activity diagram merupakan diagram yang mendefinisikan alur diagram aktivitas dari *user* dan sistem aplikasi, aktivitas tersebut akan dipisahkan dengan beberapa tabel yang memisahkan antara *user* dan sistem. Adapun *activity diagram* digambarkan di **Gambar 2**.

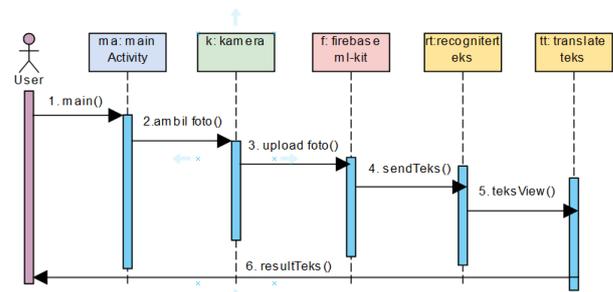


Gambar 2. Activity Diagram

d. Squence Diagram

Sequence Diagram merupakan diagram yang mendefinisikan bagaimana suatu aktivitas akan dilakukan, seperti pesan apa yang dikirimkan

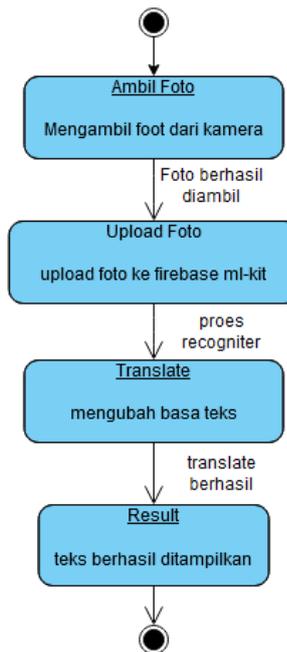
dan kapan waktu pengirimannya. Adapun *sequence diagram* digambarkan di **Gambar 3**.



Gambar 3. Squence Diagram

e. Statemachine Diagram

State machine diagram adalah teknik penggambaran diagram untuk menggambarkan aktivitas sebuah sistem. Hal ini ditujukan untuk membantu analis, perancangan dan pengembangan dilakukan untuk memahami perilaku objek pada sistem. Adapun *statemachine diagram* digambarkan di **Gambar 4**.



Gambar 4. Statemachine Diagram

3.3 Evaluasi Prototyping

Hasil evaluasi dari perancangan prototyping ini sudah sesuai yang diinginkan, oleh karena itu akan dilanjutkan ke tahap keempat.

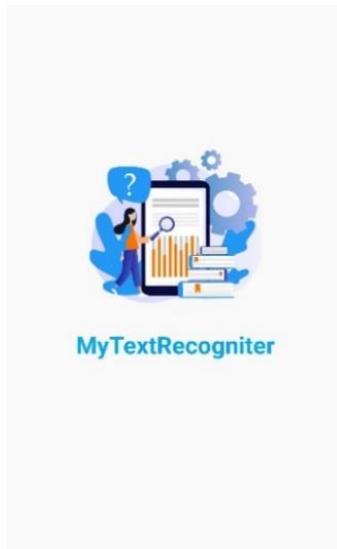
3.4 Pengkodean Sistem

Aplikasi ini menggunakan bahasa pemrograman *KOTLIN* untuk pembuatannya, sedangkan untuk layout/tampilan aplikasi android menggunakan *XML*, dan fitur *recogniter*

menggunakan *firebase ML-KIT*. Pengkodean dari sistem ini adalah:

a. SplashScreen

Halaman *SplashScreen* ini menampilkan gambar logo selama 2 detik yang kemudian pindah ke halaman dashboard. *Splashscreen* ditampilkan di **Gambar 5**.



Gambar 5. *SplashScreen*

b. Dashboard.

Halaman *Dashboard* atau halaman utama ini dimulai dengan membuka kamera pada *icon* kamera untuk mengambil foto. *Dashboard* ditampilkan di **Gambar 6**.



Gambar 6. Dashboard

c. Upload Foto

Mengunggah foto yang diambil dari kamera dengan menekan tombol *Translate*, yang di mana gambar akan diproses ke *teks recognizer* dan diterjemahkan. Upload Foto ditampilkan di **Gambar 7**.



Gambar 7. Upload Foto

d. Result

Teks akan ditampilkan pada *Translate Activity* yang di mana akan menampilkan teks yang sudah ditranslate dari foto yang diambil. *Result* ditampilkan di **Gambar 8**.



Gambar 8. *Result*

3.5 Pengujian Sistem

Pengujian ini menggunakan *Black Box* adalah pengujian yang di mana pengujian program dengan cara melihat kinerja aplikasinya tanpa perlu

mengetahui susunan atau struktur program (Putra, Nurdin, Rondonuwu, & Kusyadi, 2020). Pengujian ini dilakukan untuk melihat hasil sistem apakah telah sesuai atau belum. Untuk itu, user hanya bisa memasukkan (*input*) data yang benar ke dalam sistem. Hasil dari pengujian sistem ini adalah sebagai berikut. Adapun hasil pengujian dengan *black box* terdapat di **Tabel 5**.

Tabel 5. Hasil Pengujian *Black Box*

Skenario	Kasus	Harapan Hasil	Hasil Pengujian	Kesimpulan
Membuka Aplikasi	Menekan icon aplikasi	SplashScreen terbuka selama 2 detik kemudian pindah ke halaman utama	Sesuai	Normal
Membuka Kamera	Menekan tombol icon kamera	Kamera terbuka	Sesuai	Normal
Upload Foto	Menekan tombol translate	Foto akan diubah menjadi teks kemudian di translate	Sesuai	Normal
Memilih Bahasa	Menekan tombol Swip Language	Bahasa berhasil dipilih (indonesia- inggris)	Sesuai	Normal
Result	Proses sistem	Hasil teks berhasil ditampilkan	Sesuai	Normal

Untuk menghasilkan pengujian teks recognizer yang baik, penulis membuat pengujian tingkat akurasi, yang diambil dari beberapa sample uji berbagai macam tulisan atau disebut dengan *Character Accuracy (Cacc)* (Nugroho, Kharisma, & Wardhono, 2019). Untuk menghitung menghitung persentase tingkat akurasi pengenalan teks menggunakan rumus berikut:

$$Acc = \frac{\text{Jumlah kata yang dikenal dengan tepat}}{\text{Total semua kata yang di uji}} \times 100\% \text{ (pers 1)}$$

Tabel 6. Hasil Pengujian Teks

No	Sampel Uji	Hasil Translate	Kesimpulan
1	This	Ini	Sesuai
2	Money	Uang	Sesuai
3	School	Sekolah	Sesuai

No	Sampel Uji	Hasil Translate	Kesimpulan
4	Go To Home	Pergi Ke Rumah	Sesuai
5	How Are You ?	Bagaimana Kabarmu	Sesuai
6	My Name Is Pascal	Nama Saya Pacal	Sesuai
7	I Love You	Saya Cinta Anda	Sesuai
8	I Will Remember You	Aku Akan Mengingatmu	Sesuai
9	I Will Not Forget You	Aku Tidak Akan Melupakanmu	Sesuai
10	We Have To Fight Lazy To Learn	Kita Harus Melawan Malas Untuk Belajar	Sesuai
11	Books Are Repository Of Knowledge	Buku Adalah Repositori Pengetahuan	Kurang Sesuai
12	Monday, Tuesday, Wednesday, Thursday, Friday, Saturday And Sunday Are The Names Of The Day	Senin, Selasa, Rabu, Kamis, Jumat, Sabtu dan Minggu Adalah Nama Hari	Sesuai
13	Learning Is One Of The Keys To Success	Belajar Adalah Salah Satu Kunci Kesuksesan	Sesuai
14	Patience Is The Solution To The Problem	Kesabaran Adalah Solusi untuk Masalah	Sesuai

Hasil dari **Tabel 6**. Yaitu pengujian teks yang diterjemahkan terdapat beberapa teks yang kurang sesuai dengan yang diharapkan, pendeteksian teks tersebut dipengaruhi oleh *font* teks, pencahayaan pengambilan gambar dan posisi pengambilan gambar. Berdasarkan pengujian pada tabel 6 tingkat akurasi pendeteksian teks dari sampel karakter yang telah ambil yaitu:

Jumlah sampel = 14 sampel

Jumlah seluruh kata = 68 kata

Jumlah hasil yang sesuai = 62 dari jumlah seluruh kata

Selanjutnya, untuk mencari keakuratan dari pendeteksian teks (Cacc) maka menggunakan rumus pada persamaan 1 dan mendapatkan hasil sebagai berikut:

$$R = \frac{62}{68} = 0,911 \times 100\% = 91,1\% \text{ (pers 1)}$$

Jadi tingkat akurat pendeteksian teks pada sistem aplikasi ini adalah sebesar 91,1% berdasarkan perhitungan yang dilakukan. Kurangnya tingkat akurat pada sistem ini adalah

8,9% dari keseluruhan sampel uji coba yaitu terdapat satu (1) pengujian yang tidak dikenali sistem.

3.6 Evaluasi Sistem

Evaluasi sistem dapat disimpulkan dalam pengambilan teks dari gambar untuk diterjemahkan harus memperhatikan *font* yang digunakan, pencahayaan dalam mengambil gambar, dan posisi pengambilan gambar.

4 Kesimpulan dan Saran

Kesimpulan pada penelitian ini adalah sistem dapat mendeteksi teks bahasa asing (inggris) yang kemudian diubah kedalam bahasa indonesia, sistem ini menggunakan fitur dari *firebase ML-KIT* untuk pengenalan dan terjemahan teks. Perancangan sistem menggunakan metode prototyping karena lebih mudah digunakan dan efektif untuk pembuatan sistem. Aplikasi ini berguna untuk membantu masyarakat Desa Gunungendut dalam membantu memahami teks bahasa inggris yang terdapat pada gambar. Hasil perhitungan akurasi dari pendeteksian teks adalah sebesar 91,1% dan akurasi yang kurang adalah sebesar 8,9%, yang diambil dari 68 sampel kata.

Saran untuk penelitian selanjutnya adalah bahasa diubah tidak hanya satu bahasa (inggris – indonesia) melainkan beberapa bahasa, bahkan bahasa daerah seperti sunda, jawa dan lainnya, sehingga dapat memberi kemudahan dalam memahami bahasa daerah tersebut.

Referensi

- Alfaris, H. B., Anam, C., & Masy'an, A. (n.d.). Implementasi Black Box Testing pada Sistem Informasi Pendaftaran Santri Berbasis Web dengan Menggunakan Php dan Mysql. *SAINTEKBU: Jurnal Sains dan Teknologi*, 23-38.
- Ernawati, S., Wati, R., & Maulana, I. (2021). Penerapan Model Fountain Untuk Pengembangan Aplikasi Text Recognition dan Text To Speech Berbasis Android Menggunakan Flutter. *Prosiding Seminar Nasional Aplikasi Sains & Teknologi (SNAST)*, 178-186.
- Fikriyya, A., & Dirgahayu, R. T. (2020). Implementasi Prototyping dalam Perancangan Sistem Informasi Sekolah Desa Pendar Foundation Yogyakarta.
- Ghifari, M. H., & Susilo, A. (2020). CORA: Aplikasi Baca Untuk Lansia Berbasis Android Menggunakan Teknologi Optical. *Jurnal Teknologi Informasi*.
- Ginting, N. B., Afrianto, Y., & Suratun. (2019). Rancang Bangun Aplikasi Sistem Kontrol Perkuliahan Menggunakan Metode Black Box. *Jurnal SIMETRIS*, 577-588.
- Nugroho, F. M., Kharisma, A. P., & Wardhono, W. S. (2019). Pengembangan Aplikasi Pembelajaran Kanji menggunakan MLKit Text Recognition, Text-to-Speech dan Kanji Alive API (Studi Kasus: Kun-Yomi pada JLPT N5-N3). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 5421-5429.
- Nur, M., Irwan, S., & Santosa, D. (2019). Identifikasi Visual Cacat Produk Menggunakan Neural Network Model Backpropagation. *Jurnal Pengembangan IT (JPIT)*, 165-169.
- Punkastyo, D. A. (2018). Perancangan Aplikasi Tutorial Jurus Dasar Beladiri Cimande Menggunakan Metode Prototype. *Jurnal Informatika Universitas Pamulang*, 87-93.
- Putra, A. P., Nurdin, N., Rondonuwu, R. V., & Kusyadi, I. (2020). Implementasi Teknik Equivalence Partitions untuk Pengujian Black Box pada Sistem Informasi DAPODIKDASMEN. *Jurnal Teknologi Sistem Informasi dan Aplikasi*, 171-177.
- Subhiyakto, E. R., & Utomo, D. W. (2017). Analisis Perancangan Aplikasi Permodelan Kebutuhan Perangkat Lunak Menggunakan Metode Prototyping. *Seminar Nasional Multi Disiplin Ilmu Unisbank (SENDI_U)*.
- Sudiatika, I. G., Dewi, K. H., & Putra, A. W. (2017). Implementasi Android Studio Sebagai Media Terapi Pengenalan Toilet Untuk Anak Autis. *Konferensi Nasional Sistem & Informatika 2017*, 102-105.
- Suendri. (2018). Implementasi Diagram UML (Unified Modelling Language) Pada Perancangan Sistem Informasi Remunerasi Dosen Dengan Database Oracle. *Jurnal Ilmu Komputer dan Informatika*, 1-9.
- Yuhistiro, K., Sulaksono, A. G., & Pratama, A. H. (2021). Implementasi Blackbox Testing Pada Aplikasi Real-Time Thermal Video Detection (Studi Kasus Deteksi Demam/Covid-19). *SMATIKA*, 16-21.
- Zulkarnain, T., Suharyanto, & Anton. (2020). Perancangan Aplikasi Optical Character Recognition Berbasis Backpropagation Pada Perangkat Mobile. *INTI NUSA MANDIRI*, 195-202.