

BAB I

PENDAHULUAN

A. Latar Belakang

Optical Character Recognition (OCR) adalah proses mendeteksi dan mengenali karakter dari gambar input dan mengubahnya menjadi *American Standard Code for Information Interchange* (ASCII) atau bentuk lain yang dapat diedit dengan mesin yang setara (Kai Ding, Zhibin Liu, Lianwen Jin dan Xinghua Zhu, 2007). Sistem OCR menginterpretasikan gambar karakter yang berupa *printed* atau *handwritten* dan mengubahnya menjadi dokumen teks yang dapat diedit yang sesuai (Jana, 2014).

Fokus utama dalam penggunaan OCR adalah untuk mendigitalisasi dokumen. Masalah ini muncul ketika dokumen berisikan sebuah *text image*, yang menghasilkan *printed text* tidak dapat disalin. Masalah ini dapat diselesaikan dengan menggunakan *machine learning* seperti *Convolutional Recurrent Neural Network*.

Dalam beberapa tahun terakhir, karena kemajuan perangkat keras, pengembangan *Artificial Intelligence* telah diciptakan, dan *Deep Learning* adalah teknologi utama negara-negara untuk bersaing dan berkembang. Gagasan utama *Deep Learning* adalah untuk menghitung dengan *Neural Network*. (Gao, 2018). Dari latar belakang tersebut diajukan sebuah aplikasi “***Optical Character***

Recognition Untuk Deteksi Kartu Mahasiswa” untuk menampilkan database mahasiswa yang masih menggunakan sistem lama.

Pada penelitian ini penulis melakukan studi kasus di Universitas Muhammadiyah Parepare dan hasil penelitian ini diharapkan dapat menggantikan sistem lama serta dapat memberikan kemudahan dan pemahaman model pengembangan sistem ini dengan baik.

B. Rumusan Masalah

Dengan mengacu pada latar belakang yang sudah di jelaskan di atas maka disusunlah rumusan masalah yang akan dibahas dalam penelitian ini adalah Bagaimana mengimplementasikan *Optical Character Recognition* (OCR) dalam menampilkan *database* mahasiswa?

C. Tujuan Penelitian

Adapun tujuan penelitian sebagai berikut :

1. Membantu memudahkan mahasiswa yang masih menggunakan sistem lama atau belum adanya *Optical Character Recognition* (OCR).
2. Aplikasi ini dibuat untuk mengimplementasikan sebuah sistem sesuai dengan latar belakang untuk memudahkan mahasiswa untuk mengoptimalkan informasi dan dijadikan sebagai media alternatif tambahan. Tujuan ini juga sebagai bentuk untuk memenuhi syarat dalam menyelesaikan tugas akhir skripsi penulis.

D. Batasan Masalah

Batasan masalah pada penelitian ini adalah :

1. Data berupa 1 kata pada gambar yang merupakan *printed text* yang didapat dari *text Recognition Data Generator*.
2. Penelitian dilakukan hanya mahasiswa Universitas Muhammadiyah Parepare angkatan 220.

E. Manfaat Penelitian

Menjelaskan manfaat dan kontribusi yang akan didapat dari hasil penelitian yaitu :

1. Manfaat dilakukannya penelitian ini adalah dapat mengenali karakter pada gambar dengan menggunakan *Optical Character Recognition* yang dapat menambah wawasan dan pengetahuan
2. Hasil dari penelitian ini diharapkan memberikan banyak manfaat kepada pengguna seperti waktu lebih fleksibel sehingga pengolahan data-data lebih terkontrol.
3. Dari penelitian ini diharapkan dapat berguna dalam pengembangan IT.

F. Sistematika Penulisan

Agar pembahasan lebih sistematis, maka tulisan ini dibuat dalam lima bab, yaitu :

1. Bab I Pendahuluan

Bab ini berisi uraian tentang apa, mengapa dan untuk apa suatu topik diteliti. Bab ini terdiri dari latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian dan sistem penulisan.

2. Bab II Tinjauan Pustaka

Bab ini terdiri dari kajian teori dan kajian hasil penelitian terdahulu.

3. Bab III Metode Penelitian

Bab ini menjelaskan detail dari cara pelaksanaan penelitian atau perancangan yang meliputi: jenis penelitian, lokasi dan waktu, alat dan bahan, rancangan sistem, teknik pengumpulan data, teknik analisis data, dan diagram alir.

4. Bab IV Hasil Penelitian dan Pembahasan

Pada bab ini dibahas hasil penelitian dan pembahasannya.

5. Bab V Penutup

Pada bab penutup mencakup kesimpulan dan saran. Keduanya dibahas secara terpisah.

BAB II

TINJAUAN PUSTAKA

A. Penelitian Terdahulu

Aplikasi OCR sebelumnya pernah dilakukan pada beberapa penelitian terdahulu. Salah satu alasan penerapan aplikasi OCR ini dilakukan di Universitas Muhammadiyah Parepare adalah untuk memudahkan dalam menangani kasus yang sedang berlangsung, beberapa contoh penelitian terdahulu yang diambil antara lain adalah:

1. Judul “*Optical Character Recognition Untuk Deteksi Pelat Mobil Dan Motor Kendaraan Pada Kampus Teknik Gowa*” oleh Yusran Mansyur (Universitas Hasanuddin Makassar). Pada penelitian ini penulis menggunakan pelat mobil dan motor kendaraan. Adanya aplikasi ini dapat memudahkan mahasiswa mendapatkan kendaraan secara akurat dan cepat.
2. Judul “Pemanfaatan Teknologi Optical Character Recognition Pada Pembacaan Kartu Tanda Penduduk” oleh Hendradito Dwi Aprillian, Hindriyanto Dwi Purnomo, S.T., MIT., Ph.D (Universitas Kristen Satya Wacana Salatiga). Tujuan dari penelitian untuk mengetahui akurasi Optical Character Recognition (OCR) pada analisis keakuratan dan kecepatan Kartu Tanda Penduduk (KTP) sebagai kelengkapan registrasi kartu kredit disistem registrasi pada bank XYZ berdasarkan efesiensi waktu dan kualitas foto (hitam putih atau berwarna, kontras foto dan ukuran pixel foto). Dengan dilakukannya analisis ini peneliti menghasilkan

manfaat bagi Bank XYZ untuk dapat mengetahui hasil data dari rata-rata keakuratan, rata-rata kecepatan dan standar deviasi pada pembacaan Kartu Tanda Penduduk pada server OCR. Dari penelitian ini mempermudah dan mempercepat proses pengolahan data diri pada masa sekarang atau masa depan dan mengoptimalkan pembacaan oleh OCR lebih maksimal kedepannya sesuai dengan keinginan development program maupun perusahaan.

3. Judul “Implementasi Optical Character Recognition Pada Sistem Operasi Android Menggunakan Library Tess-two” oleh Putra, ChristyanAdityaChandra (Universitas Brawijaya). Penelitian ini membahas tentang implementasi OCR pada sistem operasi android yang kemudian akan dikombinasikan dengan beberapa metode peningkatan citra (image enhancement). Aplikasi pada penelitian ini dikembangkan menggunakan library tess-two. Library tess-two merupakan library android yang berporos pada library tesseract yang dikembangkan oleh google. Penelitian ini akan membandingkan akurasi aplikasi OCR dengan library tess-two sebelum dan setelah penambahan sebuah metode peningkatan citra (image enhancement).
4. Judul “aplikasi e-voting berbasis mobile menggunakan optical character recognition” oleh Oka Nurul Huda, Fajri Profesio Putra, Jaroji (Politeknik Negeri Bengkalis), Aplikasi *e-voting* digunakan untuk pemilihan Badan Eksekutif Mahasiswa (BEM), dimana untuk proses verifikasi pemilih digunakan pengenalan citra karakter angka yang

terdapat pada Kartu Tanda Mahasiswa (KTM). Pengenalan karakter ini menggunakan teknik *Optical Character Recognition* (OCR). Pada proses voting untuk login ke dalam aplikasi dilakukan pengenalan nomor induk mahasiswa yang tertera pada KTM menggunakan OCR. Nomor induk tersebut akan dicocokkan dengan database mahasiswa untuk menentukan apakah memiliki hak pilih untuk selanjutnya memberikan hak pilihnya. Aplikasi e-voting ini dibangun *berbasis mobile* dan dilengkapi dengan informasi quick count. Adanya aplikasi ini dapat memudahkan pemilih dalam memberikan suara tanpa harus mendatangi tempat pemungutan suara dan membantu panitia untuk mendapatkan data hasil pemilihan secara cepat dan akurat.

B. Kajian Teori

1. Optical Character Recognition

Optical Character Recognition OCR adalah singkatan dari Optical Character Recognition. Teknologi ini memungkinkan untuk mengenali karakter secara otomatis melalui mekanisme optik. Dalam kasus manusia, mata manusia adalah mekanisme optik. Gambar yang dilihat oleh mata adalah input untuk otak. Kemampuan untuk memahami input ini bervariasi pada setiap orang menurut banyak faktor. OCR adalah teknologi yang berfungsi seperti kemampuan membaca manusia. Meskipun OCR tidak mampu bersaing dengan kemampuan membaca manusia.

OCR adalah teknologi yang memungkinkan mengonversi berbagai jenis citra seperti citra kertas yang dipindai, file PDF atau gambar yang diambil oleh

kamera digital menjadi data yang dapat diedit dan dicari. Gambar yang diambil oleh kamera digital berbeda dari citra atau gambar yang dipindai. Secara umum proses OCR dapat dilihat pada Gambar 2.1, dengan penjelasan sebagai berikut:

a. File Input

File.input berupa file citra.digital dengan format *.bmp atau *.jpg.

b. Preprocessing

Preprocessing adalah proses yang bertujuan untuk menghilangkan bagianbagian.yang tidak diperlukan padagambar.input untuk proses.selanjutnya

c. Segmentasi

Segmentasi adalah proses membagi daerah yang ingin diamati(region) pada tiap karakter.yang.dideteksi.

d. Normalisasi

Normalisasi adalah proses merubah ukuran region tiap.karakter dan ketebalan.karakter.

e. Ekstraksi ciri

Ekstraksi ciri adalah proses untuk mengambil ciri-ciri.tertentu dari karakter.yang.diamati.

f. Recognition

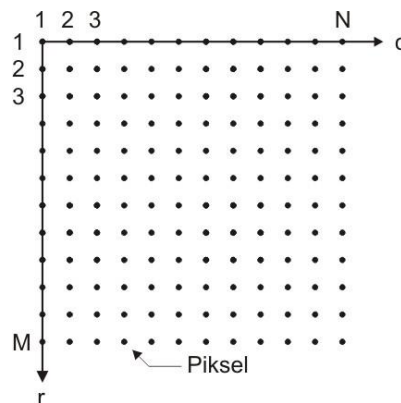
Recognition adalah proses pengenalan karakter yang dilakukan dengan cara membandingkan ciri-ciri karakter yang.diperoleh dengan ciri-ciri karakter yang ada pada basis.data



Gambar 2. 1 Proses OCR secara umum

2. Citra

Citra adalah gambar dua dimensi yang dihasilkan dari gambar analog dua dimensi yang kontinu menjadi gambar diskrit melalui proses sampling. Citra dilambangkan dengan fungsi dua dimensi, $f(x,y)$ dimana x dan y adalah koordinat spasial. Citra berwarna terdiri dari beberapa citra monokrom, misalnya RGB terdiri dari 3 kanal dan terbentuk dari kombinasi tiga warna yaitu, merah, hijau dan bir. Citra digital digambarkan dalam bentuk matriks dimana indeks baris dan kolomnya menyatakan suatu titik pada citra yang disebut piksel dan elemen matriksnya menyatakan tingkat keabuan pada titik tersebut. Persilangan antara baris dan kolom tertentu disebut dengan piksel. Gambar/titik diskrit pada baris (x) dan kolom (y) adalah piksel $[x,y]$.



Gambar 2. 2 Koordinat Piksel

3. Pengolahan Citra

Pengolahan citra adalah pemrosesan sebuah gambar dua dimensi secara digital, proses – proses yang dapat dilakukan dalam pengolahan citra digital adalah sebagai berikut:

1. Merubah dari citra berwarna ke dalam citra abu-abu.
2. Merubah dari citra berwarna atau citra abu-abu ke dalam citra biner.

Proses ini dapat dilakukan dengan deteksi tepi atau filtering.

Tujuan dari pengolahan citra digital adalah untuk memperbaiki kualitas suatu gambar agar dapat dengan mudah dipahami oleh mata manusia dan komputer untuk mengolah informasi yang terdapat pada gambar tersebut untuk keperluan pengenalan objek secara otomatis.

4. Kartu Tanda Mahasiswa (KTM)

Kartu Tanda Mahasiswa (KTM) adalah kartu identitas yang menandakan seseorang mahasiswa sedang menempuh pendidikan disuatu perguruan tinggi. Tujuan diberikannya kartu tanda mahasiswa yaitu sebagai penanda bagi seseorang yang berstatus mahasiswa.

Kampus menerbitkan Kartu Tanda Mahasiswa (KTM) secara resmi pada awal masa studi mahasiswa yang bersangkutan, dengan memuat identitas diri, identitas studi, NPM, dan nama kampus.

Masa berlaku Kartu Tanda Mahasiswa (KTM) dimulai sejak awal masa studi hingga berakhirnya studi mahasiswa yang bersangkutan. Terdapat beberapa

elemen yang dibutuhkan didalam KTM. Bagian-bagian KTM tersebut bisa dilihat pada gambar berikut:



Gambar 2. 3 Kartu Tanda Mahasiswa

1) Proses kartu tanda mahasiswa

OCR atau Optical Character Recognition merupakan salah satu ilmu komputer yang memiliki kegunaan untuk membaca teks dari kertas atau media lainnya yang diolah menjadi bentuk gambar ke bentuk digital yang membuat komputer dapat memanipulasi data yang terdapat di gambar tersebut sebagai contoh adalah ASCII ke dalam code. Sistem dari OCR sendiri adalah kombinasi dari hardware dan software untuk mengenali karakter. OCR dapat mengenali teks dari berbagai macam font, tetapi OCR masih kesulitan untuk mengenali tulisan tangan, sehingga OCR membutuhkan kalibrasi untuk membaca font secara spesifik. Berikut proses OCR kartu tanda mahasiswa :



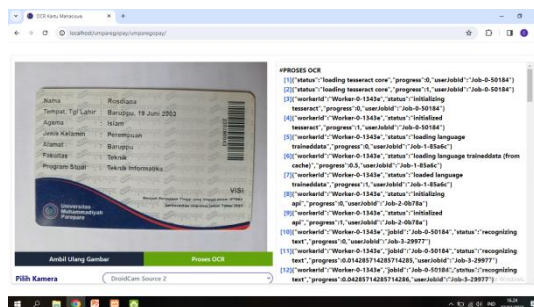
Gambar 2. 4 Halaman awal

Gambar 2.4, merupakan halaman awal dari aplikasi, dimana jika mahasiswa membuka aplikasi maka akan muncul halaman awal dari aplikasi ini. Kemudian melakukan mulai maka secara otomatis kamera akan aktif.



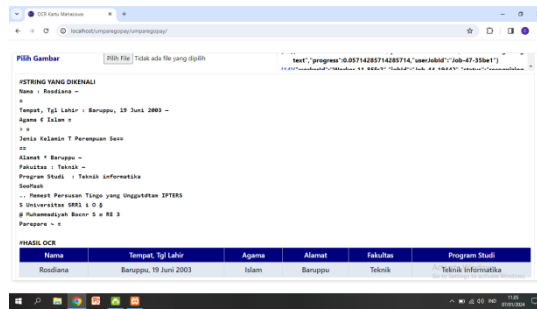
Gambar 2.5 Halaman utama aplikasi

Gambar 2.5, merupakan halaman utama aplikasi, jika mahasiswa sudah mengambil gambar maka sistem akan mengarahkan ke halaman proses ocr.



Gambar 2.6 Halaman proses ocr

Gambar 2.6 merupakan halaman proses ocr, ketika mahasiswa mengklik proses ocr yang ada pada layar aplikasi maka sistem akan melakukan proses ocr secara otomatis.



Gambar 2. 7 Halaman hasil aplikasi

Gambar 2.7 Halaman hasil aplikasi, merupakan hasil ocr yang dilakukan oleh sistem. Dimana sistem akan mendapatkan seluruh data yang ada di dalam KTM baik dari NIM sampai dengan program studi. Kemudian sistem akan menempatkan masing-masing hasil scanning pada field yang sudah disediakan.

5. Preprocessing

Perancangan sistem *preprocessing* menjelaskan alur kerja dalam OCR dalam menyiapkan citra sebagai data input yang siap diolah lebih lanjut oleh sistem. Sistem preprocessing yang digunakan pada penelitian ini adalah sebagai berikut:

a. Grayscale

Citra *grayscale* adalah citra yang hanya memiliki 1 buah kanal sehingga yang ditampilkan hanyalah nilai intensitas atau dikenal juga dengan istilah derajat keabuan. Karena jenis citra ini hanya memiliki 1 kanal saja, maka citra *grayscale* memiliki tempat penyimpanan yang lebih hemat. Jenis ini disebut juga sebagai 8- bit *image* karena untuk setiap nilai pikselnya memerlukan penyimpanan sebesar 8- bit. Rumus yang digunakan untuk mengubah citra

RGB menjadi citra *grayscale* yaitu “luma” atau “*luminance*”. Berikut adalah rumus untuk mengubah citra menjadi *grayscale*:

$$Gray = 0.299 * red + 0.587 * green + 0.114 * blue$$

b. Thresholding

Citra biner atau citra hitam putih (black and white image) adalah citra yang hanya memiliki 2 kemungkinan nilai untuk setiap pikselnya, yaitu 0 atau 1. Nilai 0 akan tampil sebagai warna hitam sedangkan nilai 1 akan tampil sebagai warna putih. Maka dari itu, jenis citra ini hanya membutuhkan 1-bit untuk menyimpan setiap nilai pada setiap pikselnya. Jenis citra ini sering digunakan untuk proses masking ataupun proses segmentasi citra. Proses thresholding digunakan untuk mengekstrak foreground (tinta) dari background (kertas) dan mengubah menjadi citra biner. Proses thresholding mengubah warna gambar menjadi citra biner (binary image) dimana ditentukan sebuah nilai level threshold kemudian piksel yang memiliki nilai level dibawah level threshold di set menjadi warna putih (1 pada nilai biner) dan nilai diatas nilai threshold di set menjadi warna hitam (1 pada nilai biner). Berikut adalah rumus biner yang digunakan:

$$g(x,y) = \begin{cases} 1 & \text{if } f(x,y) < T \\ 0 & \text{if } f(x,y) > T \end{cases}$$

Keterangan: *Gray* merupakan nilai hasil *grayscale* dan 128 adalah nilai level threshold.

6. Segmentasi

Segmentasi adalah sebuah proses yang bertujuan untuk membagi sebuah citra menjadi daerah pilihan atau mengisolasi objek dari citra secara keseluruhan, segmentasi dapat dilakukan berdasarkan tekstur, kecerahan, serta intensitas jumlah piksel. Pada proses ini akan melakukan pembagian citra menjadi dua wilayah, yaitu wilayah latar dan wilayah teks. Setelah melakukan pembagian, OCR akan melakukan proses selanjutnya hanya pada wilayah teks yang sudah terbagi. Pada penelitian ini metode segmentasi yang digunakan adalah profil proyeksi, metode ini bekerja dengan dua tahap yaitu mencari baris dari citra (horizontal) dan mencari karakter (vertical). Profil Proyeksi akan mencari garis pada teks secara vertikal dan horizontal, dimana proyeksi horizontal akan mendapatkan baris teks kemudian proyeksi vertikal akan memisahkan setiap kolom karakter. Dimana $S(N, M)$ merupakan citra biner dengan N baris dan M kolom. Profil proyeksi terdapat 2 jenis yaitu:

a. Profil Vertikal

Menjumlahkan pixel putih yang tegak lurus dengan sumbu y , yang diwakili vektor P_{ver} dengan ukuran N yang didefinisikan sebagai:

$$p_{ver}[k] = \sum_{j=1}^m [b_{kj}, k]$$

Dengan ketentuan sebagai berikut :

M = tinggi citra

$P_{ver}[k]$ = jumlah piksel pada kolom k citra

b. Profil Horizontal

Menjumlahkan pixel putih yang tegak lurus dengan sumbu x, yang diwakili vektor Phor dengan ukuran M yang didefinisikan sebagai:

$$p_{hor}[b] = \sum_{j=1}^N I[b, kj] \quad (2.4)$$

Dengan ketentuan sebagai berikut :

N = lebar citra

Phor[b] = jumlah piksel pada baris b citra

7. Ekstraksi Ciri

Ekstraksi ciri adalah proses pengukuran terhadap data yang telah di normalisasi untuk membentuk sebuah nilai fitur. Nilai fitur digunakan oleh pengklasifikasi untuk mengenali unit masukan dengan unit target keluaran dan memudahkan pengklasifikasian karena nilai ini mudah untuk dibedakan.

Ekstraksi ciri (feature extraction) dapat dikatakan sebagai proses fundamental dari sistem pengenalan karakter. Fitur/ciri adalah karakteristik unik dari suatu objek. Tujuan dari ekstraksi ciri ini adalah mendapatkan karakteristik suatu karakter yang berguna untuk membedakan antara karakter yang satu dengan lainnya. Ciri yang baik adalah ciri yang memiliki daya pembeda tinggi, sehingga proses klasifikasi pada ciri karakter bisa mendapatkan akurasi yang baik. Pada penelitian ini, metode yang digunakan untuk ekstraksi ciri adalah metode zoning:

8. UML (Unified Modeling Language)

Sora (2015), *Unified Modelling Language (UML)* adalah sebuah "bahasa" yang telah menjadi standar dalam industri untuk *visualisasi*, merancang dan mendokumentasikan sistem piranti lunak. *UML* menawarkan sebuah standar untuk merancang model sebuah sistem. *UML* adalah sekumpulan alat yang digunakan untuk melakukan abstraksiterhadap sebuah sistem atau perangkat lunak berbasis objek. *UML* merupakan singkatan dari *Unified Modeling Language*. *UML* juga menjadi salah satu cara untuk mempermudah pengembangan aplikasi yang berkelanjutan.

Aplikasi atau sistem yang tidak terdokumentasi biasanya dapat menghambat pengembangan karena developer harus melakukan penelusuran dan mempelajari kodeprogram *UML* juga dapat menjadi alat bantu untuk transfer ilmu tentang sistem atau aplikasi yang akan dikembangkan dari satu *developer* ke *developer* lainnya. Tidak hanya antar *developer* terhadap orang bisnis dan siapapun dapat memahami sebuah sistem dengan adanya *UML*.

UML diciptakan oleh *Object Management Group* yang diawali dengan versi 1.0 pada Januari 1997. Dalam pengembangan berorientasi objek ada beberapa prinsip yang harus dikenal: *Object*, *Class*, *Abstraction*, *Encapsulation*, *Inheritance* dan *Polymorphism*.

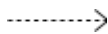
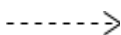
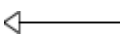
Dalam *UML* sendiri terdapat beberapa diagram yaitu :

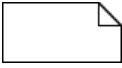
a. Use Case Diagram

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan

“bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-create sebuah daftar belanja, dan sebagainya. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. *Use case* adalah suatu urutan interaksi yang saling berkaitan antara sistem dan aktor. *Use case* dijalankan melalui cara menggambarkan tipe interaksi antara user suatu program (sistem) dengan sistemnya sendiri.

Tabel 2. 1 *Use Case Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

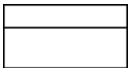
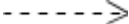
NO	GAMBAR	NAMA	KETERANGAN
7		<i>Sistem</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		Use Case	<i>Deskripsi</i> dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi.

b. Class diagram

Class diagram adalah visual dari struktur sistem program pada jenis-jenis yang di bentuk. *Class diagram* merupakan alur jalannya database pada sebuah sistem. *Class diagram* merupakan penjelasan proses database dalam suatu program. Dalam sebuah laporan sistem maka class diagram ini wajib ada.

Tabel 2. 2 Simbol *Class Diagram*

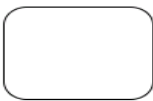
NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.

NO	GAMBAR	NAMA	KETERANGAN
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu <i>actor</i>
5		<i>Realization</i>	Operasi yang benar-banar dilakukan oleh suatu <i>object</i> .
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri.
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.

c. Activity diagram

Adalah sesuatu yang menjelaskan tentang alir kegiatan dalam program yang sedang dirancang, bagaimana proses alir berawal, keputusan yang mungkin terjadi, dan bagaimana sistem akan berakhir.

Tabel 2. 3 Simbol Activity Diagram

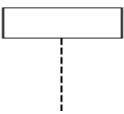
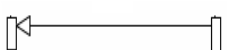
NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu Aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.

NO	GAMBAR	NAMA	KETERANGAN
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang berubah menjadi beberapa aliran.

d. Sequence Diagram

adalah suatu diagram yang menjelaskan interaksi objek dan menunjukkan (memberi tanda atau petunjuk) komunikasi diantara objek-objek tersebut.

Tabel 2. 4 Simbol *Sequence Diagram*

NO	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi tentang aktifitas yang terjadi
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi tentang aktifitas yang terjadi

9. Flowchart

Arifianto, Rahmad (2014), *Flowchart* (diagram alir) dapat digunakan sebagai alternative untuk menyajikan algoritma. *Flowchart* adalah bentuk penyajian grafis yang menggambarkan solusi langkah demi langkah terhadap suatu masalah.

Tabel 2. 5 Simbol *Standar Flowchart*

Simbol	Nama dan Fungsi
	Terminator (Tanda mulai atau selesai), menyatakan titik awal dan titik akhir diagram alir.
	Proses (Tanda komputasi), menyatakan sebarang proses.
	Keputusan (Pengambilan keputusan), digunakan untuk pengambilan keputusan. Dalam hal ini yang ada didalam symbol ini berupa suatu pertanyaan yang jawabannya berupa Ya atau Tidak.
	Proses Terdefinisi (Tanda prosedur atau fungsi subalgoritma), menyatakan prosedur lain yang telah didiagramalirkan.
	Konektor (Tanda penghubung), digunakan untuk menghubungkan ke berbagai bagian dalam alir diagram.

10. Ascii (*American Standard Code for Information Interchange*)

(ASCII) singkatan dari *American Standard Code for Information Interchange* atau **Kode Standar Amerika untuk Pertukaran Informasi** adalah standar pengodean karakter untuk alat komunikasi. Kode ASCII mewakili teks dalam komputer, peralatan telekomunikasi, dan perangkat lainnya. Kebanyakan skema pengkodean karakter modern didasarkan pada ASCII, meskipun mereka mendukung banyak karakter tambahan.

ASCII dikembangkan dari kode telegraf. Penggunaan komersial pertamanya adalah sebagai kode teleprinter tujuh-bit yang dipromosikan oleh layanan data Bell. Pekerjaan pada standar ASCII dimulai pada tanggal 6 Oktober

1960, dengan pertemuan pertama dari American Standards Association (ASA) (sekarang American National Standards Institute atau ANSI) X3.2 sub-komite.

Jumlah kode ASCII adalah 255 kode. Kode ASCII 0..127 merupakan kode ASCII untuk manipulasi teks; sedangkan kode ASCII 128..255 merupakan kode ASCII untuk manipulasi grafik. Kode ASCII sendiri dapat dikelompokkan lagi kedalam beberapa bagian:

- Kode yang tidak terlihat simbolnya seperti Kode 10(Line Feed), 13(Carriage Return), 8(Tab), 32(Space)
- Kode yang terlihat simbolnya seperti abjad (A..Z), numerik (0..9), karakter khusus (~!@#\$%^&*()_+?:'"})
- Kode yang tidak ada di keyboard namun dapat ditampilkan. Kode ini umumnya untuk kode-kode grafik.

Dalam pengkodean kode ASCII memanfaatkan 8 bit. Pada saat ini kode ASCII telah tergantikan oleh kode UNICODE (Universal Code). UNICODE dalam pengkodeannya memanfaatkan 16 bit sehingga memungkinkan untuk menyimpan kode-kode lainnya seperti kode bahasa Jepang, Cina, Thailand dan sebagainya.

Pada papan keyboard, aktifkan numlock, tekan tombol ALT secara bersamaan dengan kode karakter maka akan dihasilkan karakter tertentu. Misalnya: ALT + 44 maka akan muncul karakter koma (,). Mengetahui kode-kode ASCII sangat bermanfaat misalnya untuk membuat karakter-karakter tertentu yang tidak ada di keyboard.

Tabel berikut berisi karakter-karakter ASCII . Dalam sistem operasi Windows dan MS-DOS, pengguna dapat menggunakan karakter ASCII dengan menekan tombol Alt+[nomor nilai ANSI (desimal)]. Sebagai contoh, tekan kombinasi tombol **Alt+87** untuk karakter huruf latin "W" kapital.

Tabel 2. 6 Simbol Karakter ASCII

Karakter	Nilai Unicode (heksadesimal)	Nilai ANSI ASCII (desimal)	Keterangan
NUL	0000	0	Null (tidak tampak)
SOH	0001	1	Start of heading (tidak tampak)
STX	0002	2	Start of text (tidak tampak)
ETX	0003	3	End of text (tidak tampak)
EOT	0004	4	End of transmission (tidak tampak)
ENQ	0005	5	Enquiry (tidak tampak)
ACK	0006	6	Acknowledge (tidak tampak)
BEL	0007	7	Bell (tidak tampak)
BS	0008	8	Menghapus satu karakter di belakang kursor (Backspace)

HT	0009	9	Horizontal tabulation
LF	000A	10	Pergantian baris (Line feed)
VT	000B	11	Tabulasi vertikal
FF	000C	12	Pergantian baris (Form feed)
CR	000D	13	Pergantian baris (carriage return)
SO	000E	14	Shift out (tidak tampak)
SI	000F	15	Shift in (tidak tampak)
DLE	0010	16	Data link escape (tidak tampak)
DC1	0011	17	Device control 1 (tidak tampak)
DC2	0012	18	Device control 2 (tidak tampak)
DC3	0013	19	Device control 3 (tidak tampak)
DC4	0014	20	Device control 4 (tidak tampak)
NAK	0015	21	Negative acknowledge (tidak tampak)
SYN	0016	22	Synchronous idle (tidak tampak)

ETB	0017	23	End of transmission block (tidak tampak)
CAN	0018	24	Cancel (tidak tampak)
EM	0019	25	End of medium (tidak tampak)
SUB	001A	26	Substitute (tidak tampak)
ESC	001B	27	Escape (tidak tampak)
FS	001C	28	File separator
GS	001D	29	Group separator
RS	001E	30	Record separator
US	001F	31	Unit separator
SP	0020	32	Spasi
!	0021	33	Tanda seru (exclamation)
"	0022	34	Tanda kutip dua
#	0023	35	Tanda pagar (kres)
\$	0024	36	Tanda mata uang dolar

%	0025	37	Tanda persen
&	0026	38	Karakter ampersan (&)
'	0027	39	Karakter Apostrof
(	0028	40	Tanda kurung buka
)	0029	41	Tanda kurung tutup
*	002A	42	Karakter asteris (bintang)
+	002B	43	Tanda tambah (plus)
,	002C	44	Karakter koma
-	002D	45	Karakter hyphen (strip)
.	002E	46	Tanda titik
/	002F	47	Garis miring (<i>slash</i>)
0	0030	48	Angka nol
1	0031	49	Angka satu
2	0032	50	Angka dua

3	0033	51	Angka tiga
4	0034	52	Angka empat
5	0035	53	Angka lima
6	0036	54	Angka enam
7	0037	55	Angka tujuh
8	0038	56	Angka delapan
9	0039	57	Angka sembilan
:	003A	58	Tanda titik dua
;	003B	59	Tanda titik koma
<	003C	60	Tanda lebih kecil
=	003D	61	Tanda sama dengan
>	003E	62	Tanda lebih besar
?	003F	63	Tanda tanya
@	0040	64	A keong (@)

A	0041	65	Huruf latin A kapital
B	0042	66	Huruf latin B kapital
C	0043	67	Huruf latin C kapital
D	0044	68	Huruf latin D kapital
E	0045	69	Huruf latin E kapital
F	0046	70	Huruf latin F kapital
G	0047	71	Huruf latin G kapital
H	0048	72	Huruf latin H kapital
I	0049	73	Huruf latin I kapital
J	004A	74	Huruf latin J kapital
K	004B	75	Huruf latin K kapital
L	004C	76	Huruf latin L kapital
M	004D	77	Huruf latin M kapital
N	004E	78	Huruf latin N kapital

O	004F	79	Huruf latin O kapital
P	0050	80	Huruf latin P kapital
Q	0051	81	Huruf latin Q kapital
R	0052	82	Huruf latin R kapital
S	0053	83	Huruf latin S kapital
T	0054	84	Huruf latin T kapital
U	0055	85	Huruf latin U kapital
V	0056	86	Huruf latin V kapital
W	0057	87	Huruf latin W kapital
X	0058	88	Huruf latin X kapital
Y	0059	89	Huruf latin Y kapital
Z	005A	90	Huruf latin Z kapital
[	005B	91	Kurung siku kiri
\	005C	92	Garis miring terbalik (<i>backslash</i>)

]	005D	93	Kurung sikur kanan
^	005E	94	Tanda pangkat
_	005F	95	Garis bawah (underscore)
`	0060	96	Tanda petik satu
A	0061	97	Huruf latin a kecil
B	0062	98	Huruf latin b kecil
C	0063	99	Huruf latin c kecil
D	0064	100	Huruf latin d kecil
E	0065	101	Huruf latin e kecil
F	0066	102	Huruf latin f kecil
G	0067	103	Huruf latin g kecil
H	0068	104	Huruf latin h kecil
I	0069	105	Huruf latin i kecil
J	006A	106	Huruf latin j kecil

K	006B	107	Huruf latin k kecil
L	006C	108	Huruf latin l kecil
M	006D	109	Huruf latin m kecil
N	006E	110	Huruf latin n kecil
O	006F	111	Huruf latin o kecil
P	0070	112	Huruf latin p kecil
Q	0071	113	Huruf latin q kecil
R	0072	114	Huruf latin r kecil
S	0073	115	Huruf latin s kecil
T	0074	116	Huruf latin t kecil
U	0075	117	Huruf latin u kecil
V	0076	118	Huruf latin v kecil
W	0077	119	Huruf latin w kecil
X	0078	120	Huruf latin x kecil

Y	0079	121	Huruf latin y kecil
Z	007A	122	Huruf latin z kecil
{	007B	123	Kurung kurawal buka
	007C	124	Garis vertikal (pipa)
}	007D	125	Kurung kurawal tutup
~	007E	126	Karakter gelombang (tilde)
DEL	007F	127	Delete
	0080	128	Dicadangkan
	0081	129	Dicadangkan
	0082	130	Dicadangkan
	0083	131	Dicadangkan
IND	0084	132	Index
NEL	0085	133	Next line
SSA	0086	134	Start of selected area

ESA	0087	135	End of selected area
	0088	136	Character tabulation set
	0089	137	Character tabulation with justification
	008A	138	Line tabulation set
PLD	008B	139	Partial line down
PLU	008C	140	Partial line up
	008D	141	Reverse line feed
SS2	008E	142	Single shift two
SS3	008F	143	Single shift three
DCS	0090	144	Device control string
PU1	0091	145	Private use one
PU2	0092	146	Private use two
STS	0093	147	Set transmit state
CCH	0094	148	Cancel character

MW	0095	149	Message waiting
	0096	150	Start of guarded area
	0097	151	End of guarded area
	0098	152	Start of string
	0099	153	Dicadangkan
	009A	154	Single character introducer
CSI	009B	155	Control sequence introducer
ST	009C	156	String terminator
OSC	009D	157	Operating system command
PM	009E	158	Privacy message
APC	009F	158	Application program command
	00A0	160	Spasi yang bukan pemisah kata
ı	00A1	161	Tanda seru terbalik
¢	00A2	162	Tanda sen (Cent)

£	00A3	163	Tanda Poundsterling
¤	00A4	164	Tanda mata uang (<i>Currency</i>)
¥	00A5	165	Tanda Yen
¦	00A6	166	Garis tegak putus-putus (<i>broken bar</i>)
§	00A7	167	Section sign
¨	00A8	168	Diaeresis
©	00A9	169	Tanda hak cipta (Copyright)
ª	00AA	170	Feminine ordinal indicator
«	00AB	171	Left-pointing double angle quotation mark
¬	00AC	172	Not sign
	00AD	173	Tanda strip (<i>hyphen</i>)
®	00AE	174	Tanda merk terdaftar
—	00AF	175	Macron
°	00B0	176	Tanda derajat

±	00B1	177	Tanda kurang lebih (plus-minus)
²	00B2	178	Tanda kuadrat (pangkat dua)
³	00B3	179	Tanda kubik (pangkat tiga)
´	00B4	180	Acute accent
µ	00B5	181	Micro sign
¶	00B6	182	Pilcrow sign
·	00B7	183	Middle dot

11. Java

Bahasa Java diciptakan oleh James Gosling dan Patrick Naughton dalam suatu proyek dari SUN Microsystem. Pada mulanya ingin diberi nama OAK dari pohon yang terdapat pada kantor James Gosling, tetapi kata OAK telah ada pada sun microsystem, maka diberi nama java (terinspirasi minum kopi). Browser pertama yang dapat membaca java adalah Hot Java. Setelah Browser Netscape dari perusahaan Netscape Navigator dan IE dari perusahaan Microsoft Inc dapat membaca script java, maka bahasa java semakin populer.

Vendor-vendor lain, seperti IBM, Oracle, Symantec, Inprise (Borland Inc), dan perusahaan-perusahaan mobile, seperti Nokia, Sony Ericsson, Motorola, dan Samsung juga mengadopsi teknologi java.

12. Perangkat Lunak Pendukung

1) MySQL

Banyak situs web dinamis memerlukan database backend. Database dapat berisi informasi yang ditampilkan dari halaman web kepada pengguna, atau bertujuan dari database mungkin untuk menyimpan informasi yang diberikan oleh pengguna. Dalam beberapa aplikasi, database keduanya menyediakan informasi yang tersedia dan menyimpan informasi baru.

MySQL adalah database yang paling populer untuk digunakan dalam situs web, dikembangkan menjadi cepat dan kecil, khususnya untuk situs web. MySQL sangat populer untuk digunakan dengan situs web yang ditulis dalam bahasa pemrograman PHP, PHP dan MySQL bekerja sama dengan baik.

- a) **Relational Database System.** Seperti halnya software database lain yang ada di pasaran, MySQL termasuk RDBMS.
- b) **Arsitektur Client-Server.** MySQL memiliki arsitektur client-server dimana server database MySQL terinstall di server. Client MySQL dapat berada di komputer yang sama dengan server, dan dapat juga di komputer lain yang berkomunikasi dengan server melalui jaringan bahkan internet.
- c) **Mengenal perintah SQL standar.** SQL (*Structured Query Language*) merupakan suatu bahasa standar yang berlaku di hampir semua software database. MySQL mendukung SQL versi SQL:2003.
- d) **Mendukung Sub Select.** Mulai versi 4.1 MySQL telah mendukung select dalam select (sub select).
- e) **Mendukung Views.** MySQL mendukung views sejak versi 5.0.

- f) **Mendukung *Stored Prosedured (SP)***. MySQL mendukung SP sejak versi 5.0.
- g) **Mendukung *Triggers***. MySQL mendukung trigger pada versi 5.0 namun masih terbatas. Pengembang MySQL berjanji akan meningkatkan kemampuan trigger pada versi 5.1.
- h) Mendukung *replication*.
- i) Mendukung transaksi.
- j) Mendukung *foreign key*.
- k) Tersedia fungsi GIS
- l) Free (bebas didownload).
- m) Stabil dan tangguh.
- n) Fleksibel dengan berbagai pemrograman
- o) Security yang baik.
- p) Dukungan dari banyak komunitas.
- q) Perkembangan software yang cukup cepat.

2) XAMPP

XAMPP adalah kit all-in-one yang populer dalam menginstal apache, MySQL, dan PHP dalam satu prosedur. XAMPP juga menginstal phpMyAdmin, aplikasi web yang dapat digunakan untuk mengelola database MySQL.

Menurut situs web XAMPP, XAMPP dimaksudkan sebagai lingkungan pengembangan pada komputer lokal. Sebagai lingkungan pengembangan, XAMPP dikonfigurasi untuk menjadi seterbuka mungkin. XAMPP tidak

dimaksudkan untuk penggunaan produksi yang tidak aman sebagai lingkungan produksi. Berikut yang penting dari XAMPP:

- a) htdoc, merupakan folder untuk meletakkan file yang akan dijalankan, seperti file Php, HTML, dan script lainnya.
- b) phpMyAdmin, merupakan bagian untuk mengelola database MySQL yang ada pada komputer.
- c) Control Panel, berfungsi untuk mengelola layanan (service) XAMPP, seperti start service (mulai) atau stop service (berhenti).

XAMPP adalah singkatan dari setiap huruf, yaitu:

1. X: Program ini dapat dijalankan di banyak sistem operasi, seperti Windows, Linux, Mac OS, dan Solaris.
2. A: Apache, server aplikasi web. Tugas utama apache adalah untuk menghasilkan halaman web yang benar kepada pengguna terhadap kode Php yang sudah dituliskan oleh pembuat halaman web.
3. M: MySQL, server aplikasi database. SQL (Structured Query Language) merupakan bahasa terstruktur yang difungsikan untuk mengolah database. MySQL dapat digunakan untuk membuat, mengelola database dan isinya.
4. P: Php, bahasa pemrograman web. Bahasa pemrograman Php (Hypertext Preprocessor) adalah bahasa pemrograman untuk membuat web berbasis server-side scripting. Php digunakan untuk membuat halaman web dinamis.
5. P: Perl, bahasa pemrograman untuk semua tujuan, pertama kali dikembangkan oleh Larry Wall, mesin Unix. Perl dirilis pertama kali

tanggal 18 Desember 1987 yang ditandai dengan keluarnya Perl 1. Pada versi-versi selanjutnya, Perl juga tersedia untuk berbagai sistem operasi Unix (SunOS, Linux, BSD, HP-UX), juga tersedia untuk sistem operasi seperti DOS, Windows, PowerPC, BeOS, VMS, EBCDIC, dan PocketPC.

13. Pengujian Sistem

Pengujian sistem adalah proses pemeriksaan atau evaluasi sistem atau komponen sistem secara manual atau otomatis untuk memverifikasi apakah sistem memenuhi kebutuhan-kebutuhan yang dispesifikasikan atau mengidentifikasi perbedaan-perbedaan antara hasil yang diterapkan dengan hasil yang terjadi. Pengujian seharusnya meliputi beberapa konsep berikut.

- 1) Demonstrasi validitas perangkat lunak pada masing-masing tahap di siklus pengembangan sistem.
- 2) Penentuan validitas sistem akhir dikaitkan dengan kebutuhan pemakai.
- 3) Pemeriksa perilaku sistem dengan mengeksekusi sistem pada data sampel pengujian.

Pada dasarnya pengujian diartikan sebagai aktivitas yang dapat atau hanya dilakukan setelah pengkodean (kode program selesai). Namun, pengujian seharusnya dilakukan dalam skala lebih luas. Pengujian dapat dilakukan begitu spesifikasi kebutuhan telah dapat didefinisikan. Evaluasi terhadap spesifikasi dan perancangan juga merupakan Teknik pengujian. Kategori pengujian dapat dikategorikan menjadi dua, yaitu:

- a. Berdasarkan ketersediaan logic sistem, terdiri dari Black box testing dan White box testing.
- b. Berdasarkan arah pengujian, terdiri dari Pengujian top down dan Pengujian bottom up.

14. Pengujian *Black Box*

Konsep *Black box* digunakan untuk merepresentasikan sistem yang cara kerja didalamnya tidak tersedia untuk diinspeksi. Di dalam black box, item-item yang diuji dianggap “gelap” karena logikanya tidak diketahui, yang diketahui hanya apa yang masuk dan apa yang keluar dari black box.

Pada pengujian black box, kasus-kasus pengujian berdasarkan pada spesifikasi sistem. Rencana pengujian dapat dimulai sedini mungkin di proses pengembangan perangkat lunak. Teknik pengujian konvensional yang termasuk pengujian “black box” adalah sebagai berikut:

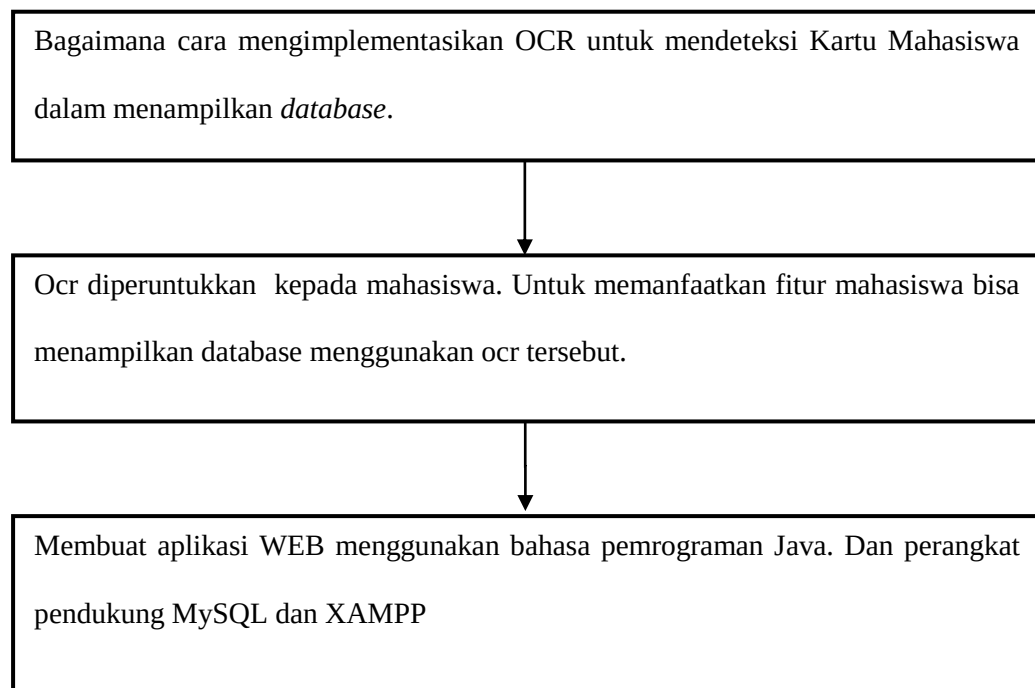
1. Graph-based testing
2. Equivalence partitioning
3. Comparison testing
4. Orthogonal array testing

Pada pengujian black box, kita mencoba beragam masukan dan memeriksa keluaran yang dihasilkan. Kita dapat mempelajari apa yang dilakukan kotak, tapi tidak mengetahui sama sekali mengenai cara konversi dilakukan. Teknik pengujian black box juga dapat digunakan untuk pengujian berbasis scenario, dimana isi dalam sistem mungkin tidak tersedia diinspeksi tapi

masukan dan keluaran yang didefinisikan dengan use case dan informasi analisis yang lain.

C. Kerangka Pikir

Untuk memahami kerangka pikir penelitian ini dapat dilihat dibawah ini.



Gambar 2. 8 Kerangka Pikir

BAB III

METODE PENELITIAN

A. Alat Dan Bahan Penelitian

Dalam penelitian ini digunakan beberapa alat dan bahan yang terdiri dari perangkat keras (Hardware) dan perangkat lunak (Software).

Perangkat keras yang terdiri dari Personal Computer (PC/Laptop), sedangkan perangkat lunak yang digunakan Sistem operasi Microsoft dan lain-lain.

Bahan lainnya yang dibutuhkan yaitu data mahasiswa dan kartu tanda mahasiswa (KTM).

B. Jenis Penelitian

Metode yang diterapkan dalam tahapan penelitian perancangan *ocr* untuk deteksi kartu mahasiswa adalah dengan menggunakan sebagai berikut :



Gambar 3. 1 Jenis Penelitian

Tahap Penelitian pada Gambar diatas, dapat dijelaskan sebagai berikut :

Tahap pertama, analisis permasalahan. Pada tahap ini dilakukan penetapan permasalahan yaitu pernyataan bersifat umum terhadap permasalahan yang diamati seperti masalah apa saja dan bagaimana cara memudahkan mahasiswa dalam mengisi data diri Kartu Tanda Mahasiswa (KTM) untuk mendapatkan keakuratan dan kecepatan proses. Tahap Kedua, melakukan pencarian dan pengumpulan literatur yang terkait seperti penelitian sebelumnya, latar belakang, teknologi dan juga teori – teori terkait seperti *Optical Character Recognition* (OCR). Tahap ketiga, Perumusan Masalah. Menguraikan permasalahan beserta solusi yang dapat membantu mengatasi permasalahan yaitu bagaimana keakuratan dan kecepatan *Optical Character Recognition* memproses foto Kartu Tanda Mahasiswa (KTM) berwarna dan *grayscale*. Tahap keempat, Rancangan Penelitian. Setelah mengetahui permasalahan dan solusi yang ditawarkan dilakukan pengolahan untuk menemukan hasil terbaik pengolahan *Optical Character Recognition* (OCR) proses keakuratan dan kecepatan pada Kartu Tanda Mahasiswa (KTM) berwarna dan *grayscale* mengambil rata – rata hasil terbaiknya. Tahap Kelima, Pengumpulan Data. Dilakukan pengumpulan Kartu Tanda Mahasiswa (KTM) berwarna dan *grayscale*. Lalu dilakukan pengujian mengunggah foto Kartu Tanda Mahasiswa (KTM) di server *Optical Character Recognition* (OCR) untuk mendapatkan hasil data Kartu Tanda Mahasiswa (KTM) yang diunggah beserta waktu prosesnya. Data – data yang sudah didapat dikumpulkan dalam bentuk table hasil pengujian. Tahap keenam, pengolahan data. Pada hasil pengujian alfa dilakukan analisis apakah hasil dari server *Optical Character Recognition* (OCR) membaca

Kartu Tanda Mahasiswa (KTM) sudah sesuai dengan yang diharapkan atau tidak Dan Didapatkan Hasil Yaitu Hasil Terbaik Dari Keakuratan Dan Kecepatan Server *Optical Character Recognition* (OCR) dalam memproses Kartu Tanda Mahasiswa (KTM). Tahap ketujuh, penyimpulan hasil. Setelah semua laporan hasil penelitian didapat, maka disimpulkan bahwa *Optical Character Recognition* (OCR) dalam membaca Kartu Tanda Mahasiswa (KTM) berwarna dan *grayscale*.

C. Waktu Dan Lokasi Penelitian

Kegiatan penelitian dilakukan di Universitas Muhammadiyah Parepare. Waktu yang dipergunakan untuk pelaksanaan penelitian ini adalah kurang lebih 3 bulan.

D. Teknik Pengumpulan Data

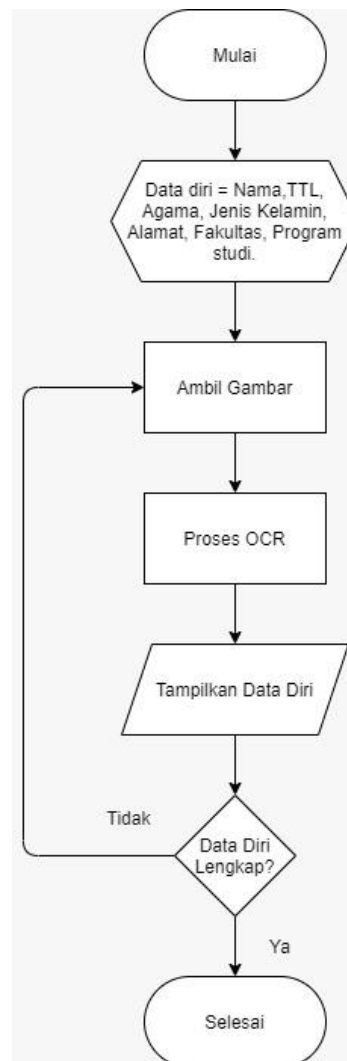
1. Studi Pustaka

Tahap ini merupakan tahap pengumpulan pengetahuan dari sumber-sumber seperti buku, dokumen atau publikasi dari berbagai sumber dan perguruan tinggi. Sumber pengetahuan tersebut dijadikan sebagai landasan teori.

2. Observasi

Observasi adalah pengamatan langsung oleh penulis dalam tahap ini penulis melakukan observasi pada universitas yang belum menggunakan *optical character recognition*

E. Desain Sistem



Gambar 3. 2 . Flowchart Aplikasi

BAB IV

HASIL DAN PEMBAHASAN

A. Hasil

Hasil dari *optical character recognition* bagaimana mahasiswa mendeteksi dan mengenali proses *ocr* dari kartu mahasiswa. Adapun hasil yang didapatkan yaitu mahasiswa dengan mudah menampilkan *database* dari *ocr* itu sendiri.

B. Pembahasan

1. Analisis Aliran data UML

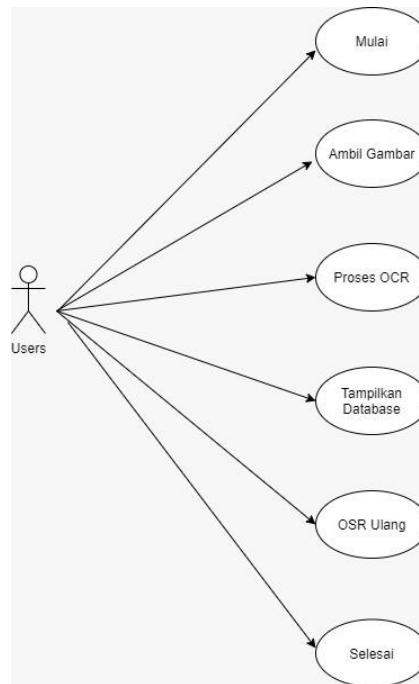
Dalam analisis sistem ini, penulis menggunakan *Use Case Diagram*, *Activity Diagram* dan *Sequence Diagram*.

a. Use Case Diagram

Use Case Diagram atau diagram *use case* merupakan pemodelan untuk memodelkan kelakuan (*behavior*) sistem informasi yang dibuat *Use Case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi. *Use case* merupakan suatu bentuk diagram yang menggambarkan fungsionalitas yang diharapkan dari sebuah sistem dilihat dari perspektif pengguna diluar sistem.

Use case juga dapat digunakan untuk meremrepresentasikan interaksi yang terjadi antara aktor dengan proses sistem yang dibuat. Pada perancangan *use case*

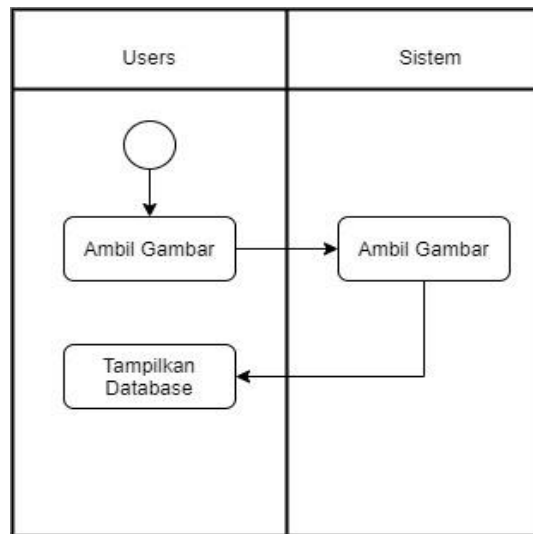
juga terdapat skenario, yaitu langkah yang menerangkan urutan kejadian antara pengguna dengan sistem diagram *use case* pada sistem.



Gambar 4.1 . Usecase Diagram

b. Activity Diagram

Activity diagram admin adalah bahasa spesifikasi standar yang dipergunakan untuk mendokumentasikan, menspesifikasikan dan membangun perangkat lunak. *Activity diagram* ini menjelaskan tentang aktifitas-aktifitas yang terjadi dalam sebuah aliran proses pada sebuah system.

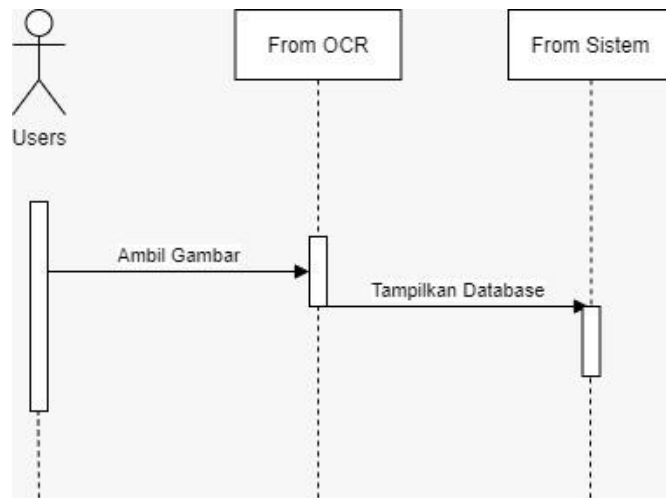


Gambar 4. 2 *Actifity Diagram*

c. Sequence Diagram

Sequence diagram atau diagram sekuen menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirim dan diterima antar objek. Proses menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah use case beserta method yang dimiliki kelas yang diinisialisasi menjadi objek yang sudah tergambar dalam class diagram.

Berdasarkan desain use case, terdapat beberapa use case yang prosesnya hampir sama satu sama lain. Untuk mempermudah pembaha-san proses dalam pembuatan diagram sekuen, berikut ini ringkasan diagram sekuen pada sistem informasi yang dikembangkan:

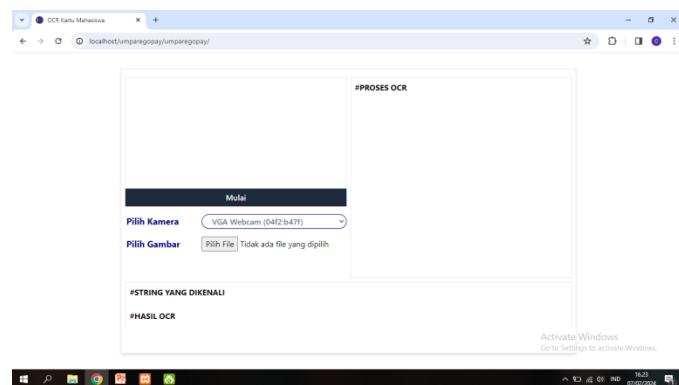


Gambar 4. 3 *Sequence Diagram*

2. Detail Aplikasi

a. Tampilan awal

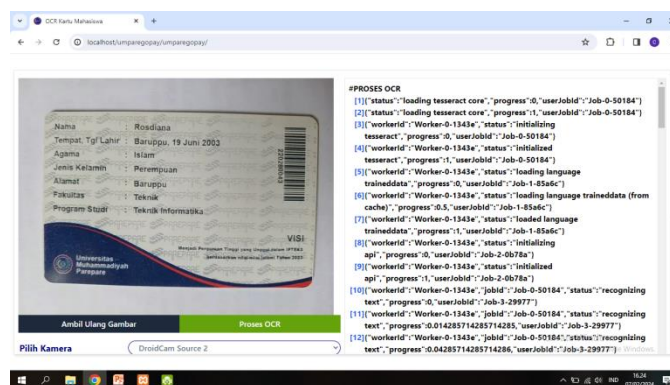
Tampilan awal merupakan halaman awal dari aplikasi, dimana jika mahasiswa membuka aplikasi maka akan muncul halaman awal dari aplikasi ini. Kemudian melakukan mulai maka secara otomatis kamera akan aktif.



Gambar 4. 4 . Tampilan awal

b. Tampilan utama

Tampilan utama merupakan halaman utama aplikasi, jika mahasiswa sudah mengambil gambar maka sistem akan mengarahkan ke halaman proses ocr.



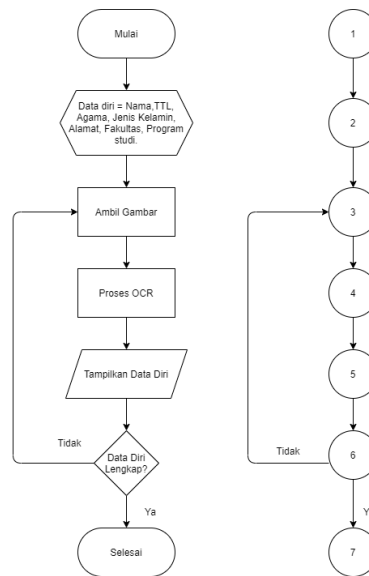
Gambar 4. 5 Tampilan utama

3. Pengujian Aplikasi

Metode pengujian perangkat lunak pada sistem aplikasi e-voting ini menggunakan pengujian *Black Box*, *White Box* dan pengujian Akurasi.

a. *White Box*.

White box testing atau pengujian kotak putih adalah salah satu metode pengujian perangkat lunak. Metode pengujian ini juga terkadang disebut juga *glass box testing* atau pengujian kotak kaca. Berbeda dengan *black box testing*, *white box testing* ini justru menguji perangkat lunak dari sisi internal tanpa memperhatikan fungsional seperti antarmuka perangkat lunak itu sendiri.



Gambar 4.6 . Tampilan Pengujian Whitebox Aplikasi

Pengujian *white box* adalah pengujian yang didasarkan pada pengecekan sistem yang dibuat, menggunakan struktur kontrol dari desain program secara procedural untuk membagi pengujian ke dalam beberapa kasus pengujian. Pemetaan *flowchart* ke dalam *flowgraph* dan proses perhitungan $V(G)$ terhadap perangkat lunak dapat dilihat pada penjelasan berikut :

Diketahui :

$$N = 7 \qquad E = 7 \qquad R = 2$$

Penyelesaian :

$$V(G) = (E - N) + 2$$

$$= (7 - 7) + 2$$

$$= 2$$

Independent Path :

Path 1 = 1 – 2 – 3 – 4 – 5 – 6 – 7

Path 2 = 1 – 2 – 3 – 4 – 5 – 6 – 3 – 4 – 5 – 6 – 7

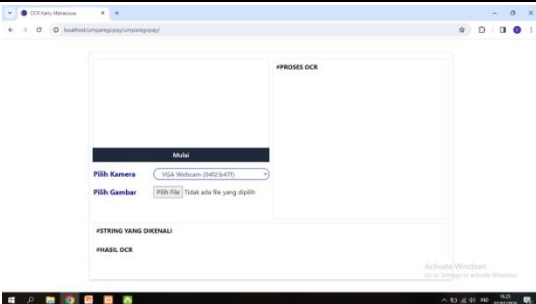
Tabel 4. 1 *White Box* Pengujian aplikasi.

	1	2	3	4	5	6	7	E-1
1		1						$1 - 1 = 0$
2			1					$1 - 1 = 0$
3				1				$1 - 1 = 0$
4					1			$1 - 1 = 0$
5						1		$1 - 1 = 0$
6					1		1	$2 - 1 = 0$
7								
SUM(E+1)								$7 - 1 = 2$

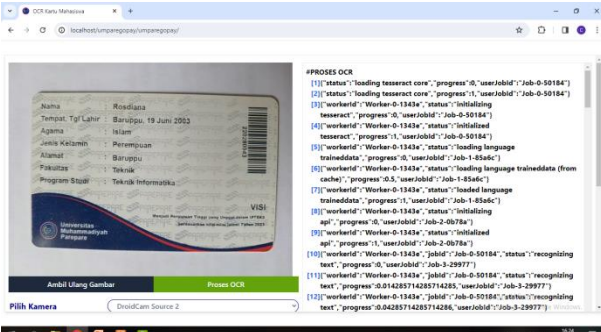
b. Pengujian *Black Box*

Pengujian *Black Box* didasarkan pada detail aplikasi, fungsi – fungsi yang ada pada aplikasi dan kesesuaian alur fungsi dengan *proses* yang diinginkan oleh pengguna atau *user*, pengujian ini tidak melihat dan menguji Logika program.

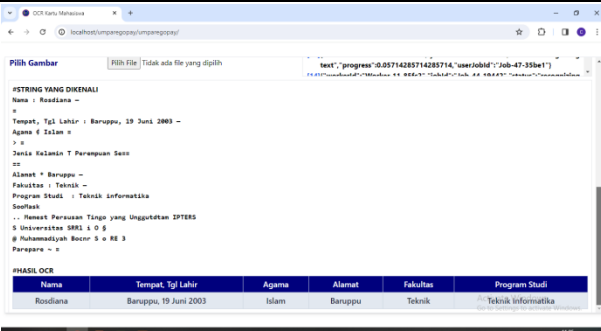
Tabel 4. 2 Pengujian Black Box Tampilan Home.

TestFaktor	Hasil	Kesimpulan
Menu Home	✓	Sukses, karena membuka aplikasi langsung muncul menu home
		

Tabel 4. 3 Pengujian *Black Box* Tampilan Home

TestFaktor	Hasil	Kesimpulan
Menu Capture gambar	✓	Sukses, ketika melakukan capture gambar maka langsung terproses ke OCR dan kemudian tampil database
		

Tabel 4. 4 . Pengujian *Black Box* Tampilan Proses OCR

TestFaktor	Hasil	Kesimpulan
Menu Tampilan Proses OCR dan Hasil	✓	Sukses, ketika melakukan proses ocr maka langsung terproses ke database
		

c. Pengujian Akurasi

Sistem ocr tersebut bisa dikatakan berhasil karena sudah beberapa kali diuji melakukan pengujian akurat dan memudahkan pengguna dalam

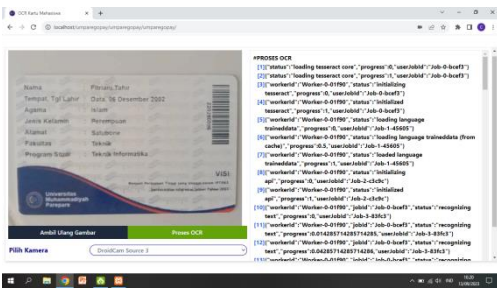
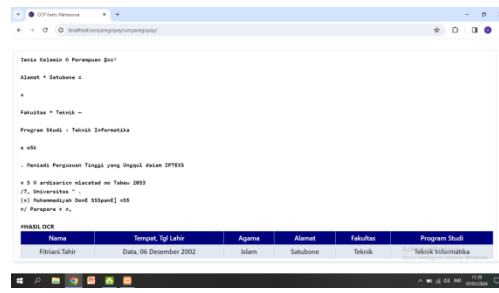
menjalankan aplikasi tersebut. Dari hasil pengujian tersebut didapatkan mahasiswa pada Universitas Muhammadiyah Parepare yang melakukan pengujian aplikasi ocr dapat dilihat pada tabel berikut:

Tabel 4. 5 Data Pengujian Mahasiswa

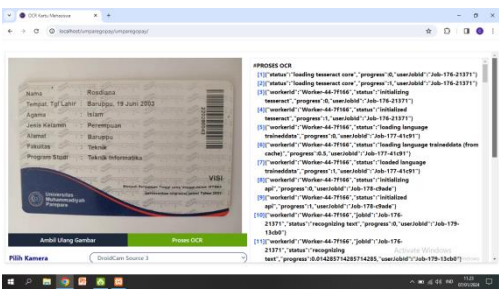
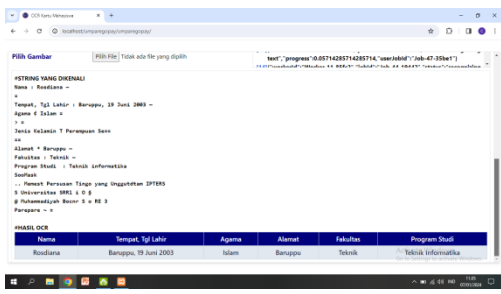
No.	Nama	Berhasil	Gagal
1.	Fitriani Tahir	✓	
2.	Puput Rahmadani	✓	
3.	Fitrawati	✓	
4.	Rosdiana	✓	
5.	Alika Febrianti	✓	

Pengujian beberapa mahasiswa dapat dilihat sebagai berikut :

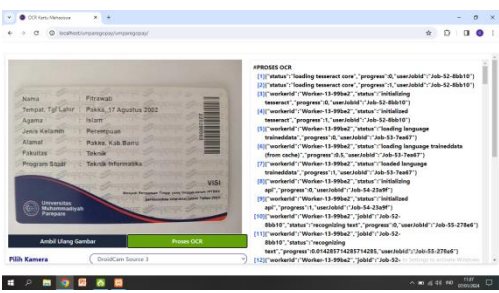
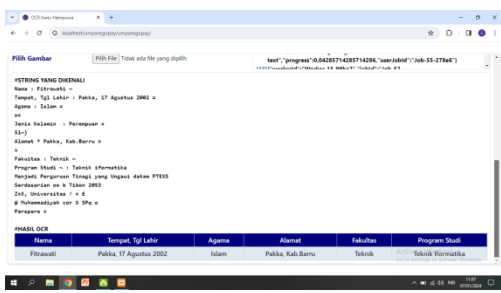
Tabel 4. 6 Proses Pengujian Mahasiswa

Pengambilan gambar			Proses dan hasil OCR		
					
Nama	Tempat, tanggal lahir	Agama	Alamat	Fakultas	Studi program
✓	✓	✓	✓	✓	✓

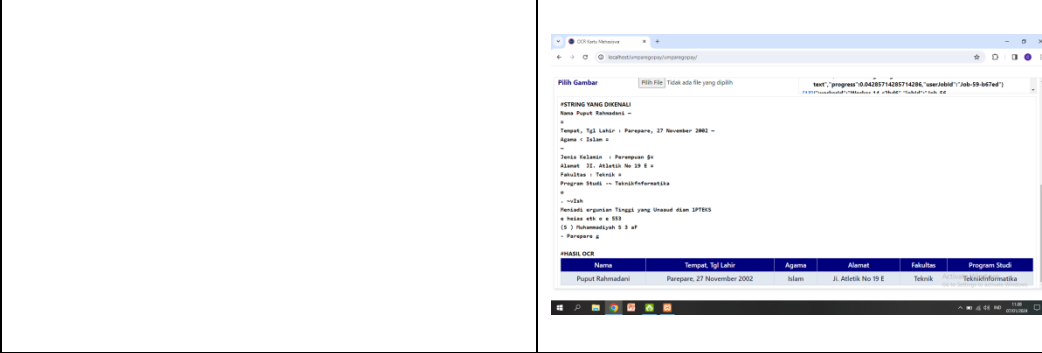
Tabel 4. 7 Proses Pengujian Mahasiswa

Pengambilan gambar			Proses dan hasil OCR		
					
Nama	Tempat, tanggal lahir	Agama	Alamat	Fakultas	Studi program
✓	✓	✓	✓	✓	✓

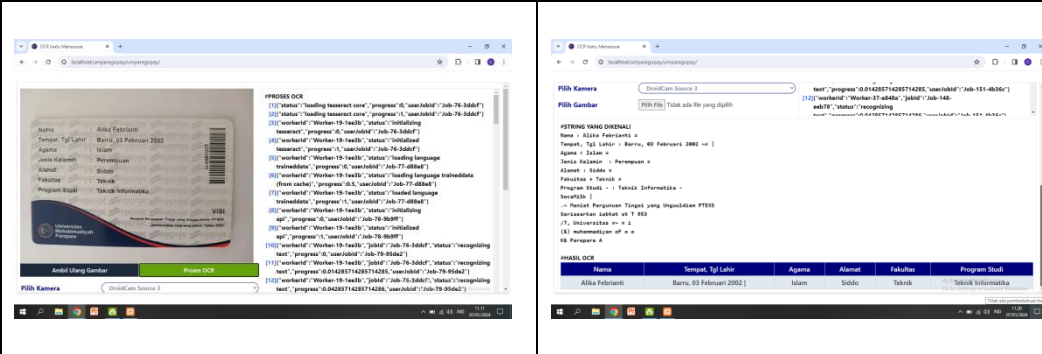
Tabel 4. 8 Proses Pengujian Mahasiswa

Pengambilan gambar			Proses dan hasil OCR		
					
Nama	Tempat, tanggal lahir	Agama	Alamat	Fakultas	Studi program
✓	✓	✓	✓	✓	✓

Tabel 4. 9 Proses Pengujian Mahasiswa

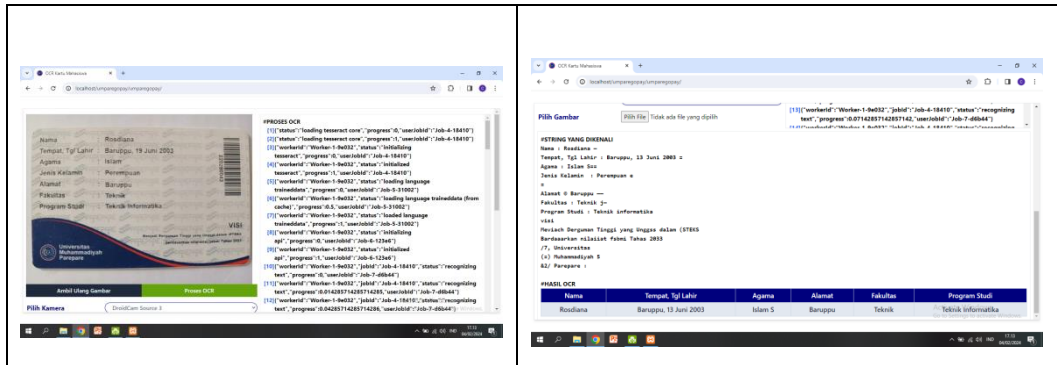
Pengambilan gambar			Proses dan hasil OCR		
					
Nama	Tempat, tanggal lahir	Agama	Alamat	Fakultas	Studi program
✓	✓	✓	✓	✓	✓

Tabel 4. 10 Proses Pengujian Mahasiswa

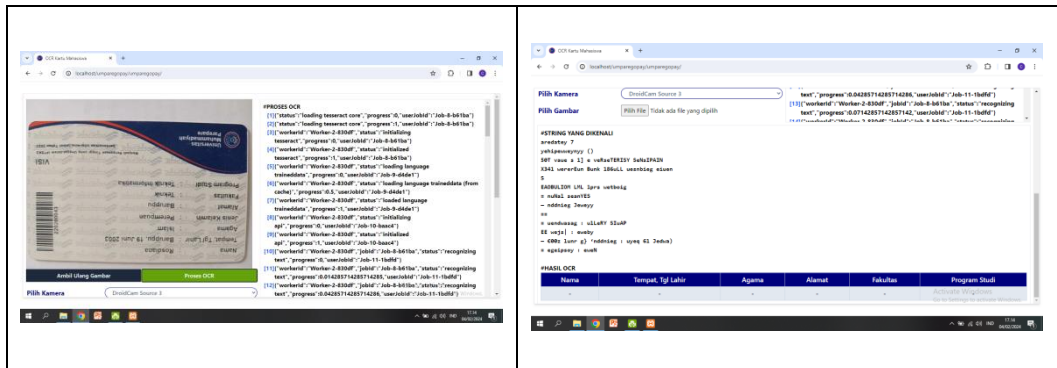
Pengambilan gambar			Proses dan hasil OCR		
					
Nama	Tempat, tanggal lahir	Agama	Alamat	Fakultas	Studi program
✓	✓	✓	✓	✓	✓

Pada pengujian diatas dilakukan pengujian menggunakan beberapa kartu mahasiswa lain dan hasil dari pengujian tersebut baik atau berjalan dengan baik. Dibawah ini dilakukan pengujian dari beberapa sudut :

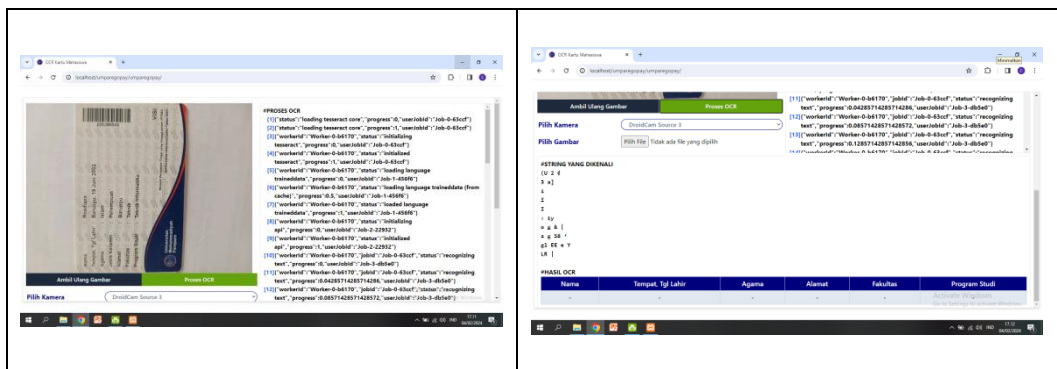
Tabel 4. 11 Pengujian Dari Beberapa Sudut

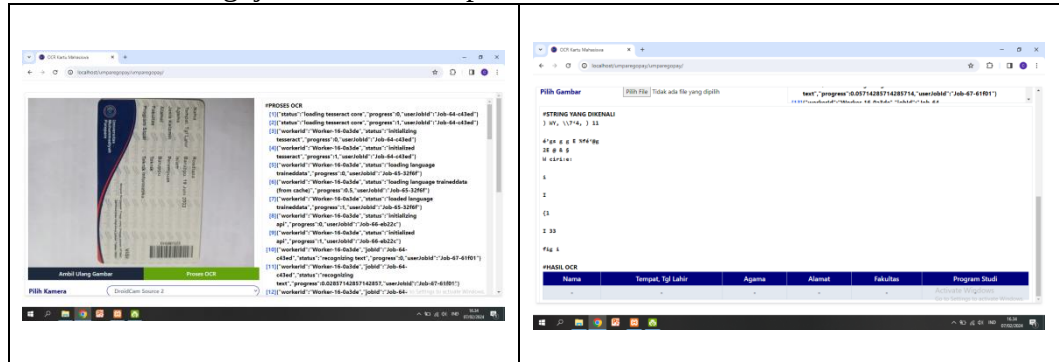


Tabel 4. 12 Pengujian Dari Beberapa Sudut



Tabel 4. 13 Pengujian Dari Beberapa Sudut

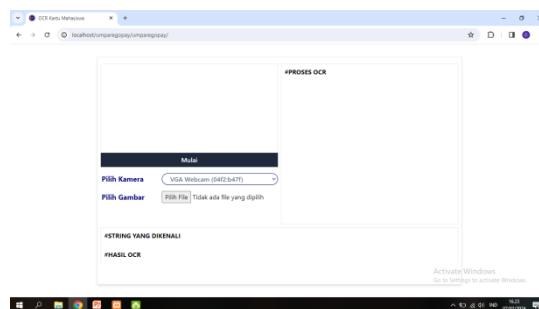


Tabel 4. 14 Pengujian Dari Beberapa Sudut

Tampilan diatas diambil dari berbagai arah dan beberapa hasil ocr diatas tidak akurat.

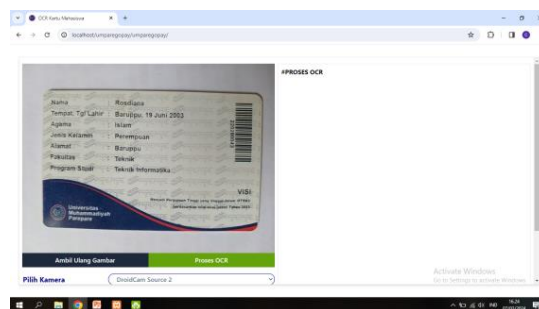
d. Proses ocr Proses kartu tanda mahasiswa

OCR atau Optical Character Recognition merupakan salah satu ilmu komputer yang memiliki kegunaan untuk membaca teks dari kertas atau media lainnya yang diolah menjadi bentuk gambar ke bentuk digital yang membuat komputer dapat memanipulasi data yang terdapat di gambar tersebut sebagai contoh adalah ASCII ke dalam code. Sistem dari OCR sendiri adalah kombinasi dari hardware dan software untuk mengenali karakter. OCR dapat mengenali teks dari berbagai macam font, tetapi OCR masih kesulitan untuk mengenali tulisan tangan, sehingga OCR membutuhkan kalibrasi untuk membaca font secara spesifik. Berikut proses OCR kartu tanda mahasiswa :

**Gambar 4. 7** Halaman awal

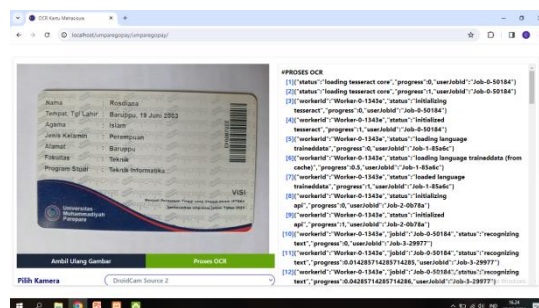
Gambar 4.7, merupakan halaman awal dari aplikasi, dimana jika mahasiswa membuka aplikasi maka akan muncul halaman awal dari aplikasi ini.

Kemudian melakukan mulai maka secara otomatis kamera akan aktif.



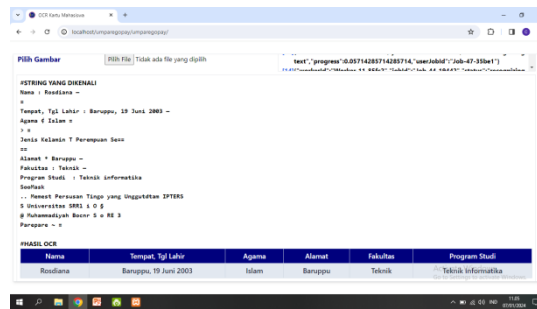
Gambar 4.8 Halaman utama aplikasi

Gambar 4.8, merupakan halaman utama aplikasi, jika mahasiswa sudah mengambil gambar maka sistem akan mengarahkan kehalaman halaman proses ocr.



Gambar 4.9 Halaman proses ocr

Gambar 4.9 merupakan halaman proses ocr, ketika mahasiswa mengklik proses ocr yang ada pada layar aplikasi maka sistem akan melakukan proses ocr secara otomatis.



Gambar 4. 10 Halaman Hasil Aplikasi

Gambar 4.10 Halaman hasil aplikasi, merupakan hasil ocr yang dilakukan oleh sistem. Dimana sistem akan mendapatkan seluruh data yang ada di dalam KTM baik dari NIM sampai dengan program studi. Kemudian sistem akan menempatkan masing-masing hasil scanning pada field yang sudah disediakan.

BAB V

PENUTUP

A. KESIMPULAN

OCR atau Optical Character Recognition merupakan salah satu ilmu komputer yang memiliki kegunaan untuk membaca teks dari kertas atau media lainnya yang diolah menjadi bentuk gambar ke bentuk digital yang membuat komputer dapat memanipulasi data yang terdapat di gambar tersebut sebagai contoh adalah ASCII ke dalam code.

Berdasarkan implementasi dan hasil pengujian yang telah dilakukan dapat disimpulkan bahwa aplikasi ini berjalan baik sesuai dengan fungsinya. Aplikasi ini berjalan pada *smartphone* berbasis *web* yang merupakan OCR kartu tanda mahasiswa (KTM) untuk membantu dan mempermudah kelancaran mahasiswa dalam menampilkan *database*.

Dari hasil pengujian aplikasi ini tidak akurat apabila posisi tidak sesuai dengan layar tampilan pada aplikasi tersebut. Pengujian dan tampilan pada aplikasi ini masih perlu dikembangkan.

B. SARAN

Dalam pengembangan aplikasi mobile ini masih jauh dari kesempurnaan karena masih memiliki banyak kekurangan. Oleh karena itu untuk pengembangan lebih lanjut mengenai aplikasi ini dapat disarankan:

1. Aplikasi dan tampilan pada aplikasi ini masih perlu dikembangkan.
2. Pengenalan OCR yang digunakan bisa langsung disorot kamera saja

DAFTAR PUSTAKA

- Gabasio, A. (2013). Comparison of Optical Character Recognition (OCR) Software. Department of Computer Science, Faculty of Engineering, LTH, Lund University. Diakses pada tanggal 26 Mei 2019
- Hartanto, S., Sugiharto, A., & Endah, S. N. (2015). Optical character recognition menggunakan algoritma template matching correlation. *Jurnal Masyarakat Informatika*, 5(9), 1-12. Diakses pada tanggal 26 Mei 2019
- M.O, A. M., Chacko, & Dhanya, P. (2015). A Comparative Study of Different Feature Extraction Techniques for Offline Malayalam Character Recognition.
- Mohammad, F., Anarase, J., Shingote, M., & Ghanwat, P. (2014). Optical Character Recognition Implementation Using Pattern Matching. *International Journal of Computer Science and Information Technologies*, 2088-2090.
- Mohammad, F., Anarase, J., Shingote, M., & Ghanwat, P. (2014). Optical character recognition implementation using pattern matching. *International Journal of Computer Science and Information Technologies*, 5(2), 2088-2090. Diakses pada tanggal 24 Mei 2019
- Nadeem, D., & Rizvi, S. (n.d.). Character Recognition using Template Matching.
- Rao, V., Sasrty, A., Chakracarthy, A., & Kalyanchakravarthi, P. (2016). Optical Character Recognition Technique Algorithms. *Journal of Theoretical and Applied Information Technology*, 83(2), 275-282.
- S Setiawan, A., Sujaini, H., & PN, A. B. (2017). IMPLEMENTASI OPTICAL CHARACTER RECOGNITION (OCR) PADA MESIN PENERJEMAH BAHASA INDONESIA KE BAHASA INGGRIS. *Jurnal Sistem dan Teknologi Informasi (JUSTIN)*, 5(2), 125-130. Diakses pada tanggal 28 Mei 2019
- Susanto, F. A. (2016). "Pemanfaatan Teknologi Optical Character Recognition (OCR) Untuk Mengenali Alfabet Yunani Berbasis Android". (Doctoral dissertation, Program Studi Teknik Informatika FTI-UKSW).
- Tan, H.-L. (1991). Hybrid feature-based and template matching optical character recognition system.