

ABSTRACT

New student registration is a mandatory activity carried out by campus organizers that aims to accommodate, screen, and select prospective students before being declared as permanent students. AMIK Labuhanbatu is one of the campus computers that continues to grow with the increasing number of new students who register each year. However, the registration process of new students at AMIK Labuhanbatu still uses manual methods that result in a lot of time wasted in carrying out the registration process, causing a reduction in time efficiency of prospective students. System design is done by the waterfall method which is a systematic and sequential model of information system development. In making a system of tools used in making the design and design of a new student registration application system, namely by using the Unified Modeling Language (UML). The new student registration application is built using Eclipse and uses the Mysql database storage media. Based on the results of the trial, it is known that the new Android-based student registration application can be used as a medium for new student registration.

Keywords: *New Student Registration, Smartphone, Application, Android.*

ABSTRAK

Pendaftaran mahasiswa baru merupakan suatu kegiatan yang wajib dilakukan pihak penyelenggara kampus yang bertujuan untuk menampung, menyaring, serta menyeleksi para calon mahasiswa sebelum dinyatakan sebagai mahasiswa tetap. AMIK Labuhanbatu adalah salah satu kampus komputer yang terus berkembang dengan bertambahnya jumlah mahasiswa baru yang mendaftar setiap tahunnya. Akan tetapi proses pendaftaran mahasiswa baru di AMIK Labuhanbatu masih menggunakan metode manual yang mengakibatkan banyaknya waktu yang terbuang dalam melakukan proses pendaftaran sehingga menyebabkan berkurangnya efisiensi waktu dari calon mahasiswa. Perancangan sistem dilakukan dengan metode waterfall yang merupakan model pengembangan sistem informasi yang sistematis dan sekuensial. Dalam pembuatan sistem alat yang digunakan dalam membuat perancangan dan desain sistem aplikasi pendaftaran mahasiswa baru yaitu dengan menggunakan Unified Modeling Language (UML). Aplikasi pendaftaran mahasiswa baru dibangun menggunakan eclipse dan menggunakan media penyimpanan database Mysql. Berdasarkan hasil uji coba, diketahui bahwa aplikasi pendaftaran mahasiswa baru berbasis Android dapat digunakan sebagai media pendaftaran mahasiswa baru.

Kata kunci: *Pendaftaran Mahasiswa Baru, Smartphone, Aplikasi, Android.*

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Pendaftaran Mahasiswa Baru (PMB) adalah suatu kegiatan yang wajib dilaksanakan oleh Perguruan Tinggi (PT), baik PT Negeri (PTN) maupun PT Swasta (PTS) untuk memperoleh mahasiswa baru. Berbagai jalur PMB ditawarkan oleh universitas. Pada Perguruan Tinggi Negeri (PTN), pemerintah menyelenggarakan seleksi bersama mahasiswa baru, seperti SNMPTN, SBMPTN, PMDK PN, dan SPAN PTKIN[1].

Namun Pengurus Pendaftaran mahasiswa baru belum dapat melakukan pengontrolan pendataannya secara maksimal terhadap proses penerimaan calon mahasiswa baru karena proses pendaftarannya masih dilakukan secara manual dengan melalui tahapan calon mahasiswa baru harus datang ke Perguruan Tinggi dan mengisi formulir pendaftaran.

Perkembangan teknologi komputer yang kian kecil tetapi mempunyai kemampuan yang besar. Demikian pula dengan kemampuan telekomunikasi yang kian pesat, hal ini dapat dilihat dengan ditemukannya sistem operasi Android pada *Smartphone*. Memicu banyak perubahan diberbagai bidang, salah satunya perguruan tinggi. Penggunaan teknologi *Smartphone* untuk melakukan pendaftaran dirasakan sangat tepat, karena teknologi ini sudah banyak digunakan dan harganya pun relatif murah. Sehingga calon mahasiswa yang telah mempunyai

smartphone dapat memaksimalkan menggunakan Android dalam melakukan proses pendaftaran[2].

Jarak yang jauh antara perguruan tinggi dengan calon mahasiswa baru, Menjadi salah satu masalah tersendiri karena terdapat pemborosan biaya saat melakukan pendaftaran mahasiswa baru.

Berdasarkan uraian diatas, maka penulis tertarik untuk membuat sebuah aplikasi berbasis *android* dalam penelitian yang berjudul; “**APLIKASI PENDAFTARAN MAHASISWA BARU BERBASIS ANDROID (STUDI KASUS : AMIK LABUHAN BATU)**” Aplikasi ini diharapkan dapat mempermudah para Calon Mahasiswa dalam Pendaftarannya .

1.2 Perumusan Masalah

Dari latar belakang permasalahan yang telah diuraikan di atas, maka dapat dirumuskan suatu permasalahan yaitu :

1. Bagaimana membangun aplikasi Pendaftaran Mahasiswa Baru Berbasis Android?
2. Bagaimana mendesain aplikasi Pendaftaran Mahasiswa Baru Berbasis Android?
3. Bagaimana melakukan pengujian terhadap aplikasi Pendaftaran Mahasiswa Baru Berbasis Android?

1.3 Batasan Masalah

Adapun batasan masalah dalam aplikasi Pendaftaran Mahasiswa Baru adalah sebagai berikut :

1. Pembangunan aplikasi Pendaftaran Mahasiswa Baru Berbasis Android dengan database *Mysql*.
2. Pembuatan aplikasi Pendaftaran Mahasiswa Baru Berbasis Android dilakukan dengan menggunakan *Eclipse*.
3. Pengaplikasian Pendaftaran Mahasiswa Baru dilakukan dengan menggunakan *smartphone* Android.

1.4 Tujuan Penelitian

Berdasarkan permasalahan yang penulis temukan, pembuatan aplikasi Pendaftaran Mahasiswa Baru berbasis android ini bertujuan untuk :

1. Membuat Aplikasi Pendaftaran Mahasiswa Baru AMIK Labuhanbatu Berbasis Android.
2. Mampu Mempermudah Calon Mahasiswa Baru AMIK Labuhanbatu dalam Pendaftaran.
3. Memenuhi salah satu syarat untuk menyelesaikan Pendidikan Diploma III pada program Studi Manajemen Informatika, AMIK Labuhanbatu.

1.5 Sistematika Penulisan

Untuk lebih memperjelas penulisan Tugas Akhir ini, Maka penulis membuat sistematika penulisan sebagai berikut :

BAB I : PENDAHULUAN

Bab ini berisikan membahas penjelasan tentang : latar belakang, rumusan masalah, batasan masalah, tujuan masalah dan sistematika penulisan.

BAB II : LANDASAN TEORI

Pada bab ini berisikan tentang teori – teori yang yang berhubungan dengan judul tugas akhir seperti, hal – hal yang berkaitan dengan *android*, bahasa yang digunakan dalam pembuatan aplikasi, database *MySql*, serta gambaran umum instansi.

BAB III : METODOLOGI PENELITIAN

Pada bab ini berisikan tentang metode pengumpulan data dan metode perancangan aplikasi pendaftaran mahasiswa baru berbasis *android* seperti perancangan *input*, perancangan proses, dan perancangan *output*.

BAB IV : HASIL DAN PEMBAHASAN

Pada bab ini berisikan tentang program yang dirancang serta hasil program. Selain itu, pada bab ini berisikan tentang penjelasan mengenai keterkaitan antara metodologi dan argumentasi dari landasan teoritis untuk memperoleh solusi pemecahan masalah.

BAB V : KESIMPULAN DAN SARAN

Bab ini mengemukakan kesimpulan dari pemecahan masalah dan memberikan saran terhadap perbaikan aplikasi yang digunakan saat ini.

BAB II

LANDASAN TEORI

2.1 Definisi Sistem.

Definisi sistem dalam kamus *Webster's Unbringed* adalah elemen-elemen yang saling berhubungan dan membentuk satu kesatuan atau organisasi. Sistem dapat juga di definisikan sebagai kumpulan dari beberapa komponen yang saling berhubungan satu dengan yang lainnya membentuk satu kesatuan untuk mencapai tujuan tertentu[3].

Menurut Lucas dalam Ladjamudin, sistem adalah suatu komponen atau variable yang terorganisir, saling berinteraksi, saling tergantung satu sama lain dan terpadu[4].

Menurut I Putu Agus Eka Pratama (2014), Sistem didefinisikan sebagai sekumpulan prosedur yang saling berkaitan dan saling berhubungan untuk melakukan suatu tugas secara bersama-sama[5].

Suatu sistem terdiri dari unsur-unsur umum yaitu masukan (*Input*), Pengolahan (*Processing*), dan Keluaran (*Output*). Unsur-unsur secara garis besar dapat digambarkan sebagai berikut :



Gambar 2.1 Alur Suatu Sistem

2.1.1 Karakteristik Sistem.

Model umumnya sebuah sistem terdiri dari *input*, *proses* dan *output*. Hal ini merupakan konsep sebuah sistem yang sangat sederhana mengingat sebuah sistem dapat mempunyai beberapa masukan, proses dan keluaran. Selain itu, sebuah sistem dapat mempunyai atau sifat-sifat tertentu, yang menceritakan bahwa hal tersebut bisa dikatakan suatu sistem. Adapun karakteristik yang dimaksud adalah sebagai berikut :

1. Komponen Sistem(komponen)

Sebuah sistem yang terdiri dari sejumlah komponen yang saling berinteraksi, yang berarti bahwa setiap bekerja bersama untuk membentuk serikat pekerja. Komponen sistem biasanya dikenal dengan subsistem. Subsistem memiliki hal sistem itu sendiri dalam fungsinya dan memiliki sistem keseluruhan.

2. Batasan Sistem (*boundary*)

Pembatasan yang membatasi sistem merupakan daerah antara sistem dengan sistem lainnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan. Menunjukkan sistem membatasi ruang lingkup sistem.

3. Lingkungan Luar Sistem (*environments*)

Lingkungan luar sistem di luar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar Sistem dapat bermanfaat serta merugikan sistem. Lingkungan eksternal yang menguntungkan merupakan energi dari sistem dan lingkungan luar yang merugikan harus ditahan dan dikendalikan, kalau tidak akan mengganggu kehidupan kelangsungan sistem.

4. Penghubung Sistem (*Interface*)

Sistem link adalah media penghubung antara subsistem lainnya. Melalui interface ini memungkinkan sumber daya mengalir dari satu subsistem ke subsistem lainnya.

5. Masukan Sistem (*Input*)

Masukan sistem adalah energi yang dimasukkan ke dalam sistem. Masukan dapat pengobatan masukan (input pemeliharaan) dan sinyal input (sinyal input). Masukan energi pemeliharaan dimasukkan sehingga sistem tersebut dapat beroperasi. Sinyal input diproses untuk mendapatkan keluaran energi.

6. Output System (*Output*)

Keluaran sistem adalah hasil dari energi dari dalam meskipun dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

Output dapat menjadi masukan bagi subsistem lain atau suprasistem.

7. Pengolahan Sistem (*proses*)

Suatu sistem dapat memiliki bagian pengolahan yang akan mengubah input menjadi output.

8. Sasaran Sistem (*objectives*)

Sebuah sasaran yang ingin dicapai untuk menentukan masukan yang diperlukan dari output sistem menjadi sistem yang dihasilkan.

2.1.2 Klasifikasi Sistem.

Sistem merupakan suatu bentuk integrasi antara suatu komponen dengan komponen lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi didalam sistem tersebut. Oleh karena itu sistem dapat diklasifikasikan dari beberapa sudut pandang yaitu diantaranya sebagai berikut:

1. Sistem Abstrak dan Sistem Fisik.

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik, misalnya sistem yang berupa pemikiran tentang hubungan antara manusia dan tuhan, sedangkan sistem fisik merupakan sistem yang ada secara fisik.

2. Sistem Alamiah dan Sistem Buatan Manusia.

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia, misalnya sistem perputaran bumi, terjadinya siang malam dan pergantian musim. Sedangkan sistem buatan manusia

manusia merupakan sistem yang melibatkan hubungan manusia dengan mesin, yang disebut dengan *human machine system*.

3. Sistem tertentu dan sistem tak tentu

Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat diprediksi. Sistem tak tentu adalah sistem yang kondisi masa depan tidak dapat diprediksi karena mengandung unsur probabilitas.

4. Sistem tertutup dan sistem terbuka.

Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak terpengaruh oleh lingkungan luarnya. Sistem ini bekerja secara otomatis tanpa cara campur tangan pihak luar. Sedangkan sistem terbuka adalah sistem yang berhubungan dan dipengaruhi oleh lingkungan luarnya. Sistem ini menerima masukan dan menghasilkan keluaran untuk subsistem lainnya.

2.2 Aplikasi

Aplikasi adalah suatu subkelas perangkat lunak komputer yang memanfaatkan kemampuan komputer langsung untuk melakukan suatu tugas yang diinginkan pengguna. Biasanya jika dibandingkan dengan perangkat lunak sistem yang dapat mengintegrasikan berbagai kemampuan komputer, tetapi tidak secara langsung akan menerapkan kemampuan tersebut untuk mengerjakan suatu tugas yang menguntungkan pengguna[6].

Pengertian aplikasi menurut para ahli adalah sebagai berikut :

- a) Menurut Jogiyanto adalah penggunaan dalam suatu komputer, instruksi (*instruction*) atau pernyataan (*statement*) yang disusun sedemikian rupa sehingga komputer dapat memproses *input* menjadi *output*.
- b) Menurut Kamus Besar Bahasa Indonesia adalah penerapan dari rancang sistem untuk mengolah data yang menggunakan aturan atau ketentuan bahasa pemrograman tertentu. Aplikasi adalah suatu program komputer yang dibuat untuk mengerjakan dan melaksanakan tugas khusus dari pengguna.
- c) Menurut Rachmad Hakim S, adalah perangkat lunak yang digunakan untuk tujuan tertentu, seperti mengolah dokumen, mengatur Windows &, permainan (game), dan sebagainya.

- d) Menurut Harip Santoso, adalah suatu kelompok file (*form, class, rePort*) yang bertujuan untuk melakukan aktivitas tertentu yang saling terkait, misalnya aplikasi *payroll*, aplikasi *fixed asset*[7].

2.3 Konsep Dasar Pendaftaran Mahasiswa Baru

2.3.1 Pendaftaran

Menurut *DEPDIKBUD* pengertian pendaftaran adalah proses, cara, perbuatan mendaftar yaitu pencatatan nama, alamat dsb dalam daftar. Jadi, pendaftaran adalah proses pencatatan identitas pendaftar kedalam sebuah media penyimpanan yang digunakan dalam proses pendaftaran[8].

2.3.2 Mahasiswa

Menurut Siswoyo (2007) mahasiswa dapat didefinisikan sebagai individu yang sedang menuntut ilmu ditingkat perguruan tinggi, baik negeri maupun swasta atau lembaga lain yang setingkat dengan perguruan tinggi. Mahasiswa dinilai memiliki tingkat intelektualitas yang tinggi, kecerdasan dalam berpikir dan perencanaan dalam bertindak. Berpikir kritis dan bertindak dengan cepat dan tepat merupakan sifat yang cenderung melekat pada diri setiap mahasiswa, yang merupakan prinsip saling melengkapi[9].

2.3.3 Penerimaan Mahasiswa Baru

Penerimaan mahasiswa baru (PMB) adalah aktifitas rutin perguruan tinggi setiap pembukaan ajaran baru. Dalam pelaksanaan PMB kini telah banyak jalur seleksi untuk melakukan pendaftaran ke perguruan tinggi. Perguruan tinggi adalah satuan pendidikan penyelenggara pendidikan tinggi, perguruan tinggi sendiri dibedakan menjadi 2, yaitu Perguruan Tinggi Negeri dan Perguruan Tinggi Swasta.

1. Perguruan Tinggi Negeri

Perguruan Tinggi Negeri adalah perguruan tinggi yang diselenggarakan oleh pemerintah. Di Indonesia, perguruan tinggi negeri dikelola oleh Kementerian Riset, Teknologi, dan Pendidikan Tinggi Republik Indonesia. Rektor perguruan tinggi negeri merupakan pejabat setingkat eselon 2 di bawah Menteri Riset, Teknologi, dan Pendidikan Tinggi Republik Indonesia ataupun kementerian lainnya.

2. Perguruan Tinggi Swasta

Perguruan Tinggi Swasta adalah perguruan tinggi yang dikelola oleh Yayasan atau lembaga nonpemerintah. Semua Universitas Swasta di Indonesia dikoordinasikan oleh badan yang disebut Koordinator Perguruan Tinggi Swasta, disingkat Kopertis. Kopertis ini bukan hanya menaungi Univeritas Swasta saja, tetapi juga Institut Swasta, dan Sekolah Tinggi Swasta lainnya.

2.4 Android.

Android adalah sebuah sistem operasi untuk perangkat mobile berbasis linux yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Awalnya, Google Inc. membeli Android Inc. yang merupakan pendatang baru yang membuat piranti lunak untuk ponsel/smartphone[10].

2.4.1 Sejarah Android

Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Awalnya, google Inc. membeli Android Inc. yang merupakan pendatang baru yang membuat perangkat lunak untuk ponsel/smartphone. Kemudian untuk mengembangkan Android dibentuklah *Open Handset Alliance*, konsorium dari 34 perusahaan piranti keras, piranti lunak dan telekomunikasi, termasuk *google, HTC, Intel, Motorola, Qualcomm, T-Mobile* dan *Nvidia*. Pada saat perilis perdana Android, 5 Nopember 2007, Android bersama Open Handset Alliance menyatakan mendukung pengembangan open source pada perangkat mobile. Di lain pihak, google merilis kode-kode Android dibawah lisensi apache, sebuah lisensi perangkat lunak dan open platform perangkat seluler. Sekitar September 2007 Google menganalkan Nexus One, salah satu jenis smartphone yang menggunakan Android sebagai system operasinya. Telepon seluler ini diproduksi oleh HTC Corporation dan tersedia di pasaran 5 Januari 2010. Pada Desember 2008, diumumkan anggota baru yang bergabung pada program kerja Android ARM *Holdings, Atheros Communications*, diproduksi oleh *Asustek computer Inc, Garmin Ltd, Softbank, Sony Ericsson, Toshoba Corp*, dan *Vodafone Group Plc*. Seiring pembentukan Open Handset Alliance, OHA mengumumkan produk perdana mereka Android, perangkat Mobile yang merupakan modifikasi kernel Linux 2.6. Sejak Android dirilis telah dilakukan berbagai pembaharuan berupa perbaikan bug dan penambahan fitur baru. Pada masa saat ini sebagian besar vendor-vendor smartphone sudah memproduksi smartphone berbasis Android, vendor-vendor itu antara lain *Dell, Nexus, SciPhone, WayteQ, Sony Ericsson, LG, Acer, Philips, TMobile, Nexian, IMO, Asus* dan masih banyak lagi vendor smartphone di dunia yang memproduksi Android. Hal ini, karena Android itu adalah sistem operasi yang open source sehingga bebas didistribusikan dan dipakai oleh vendor manapun. Tidak hanya menjadi sistem operasi di smartphone, saat ini Android menjadi pesaing utama dari Apple pada system operasi Tablet PC. Pesatnya pertumbuhan Android selain faktor yang disebutkan di atas adalah karena Android itu sendiri adalah Platform yang sangat lengkap baik itu system operasinya, Aplikasi dan Tool pengembangan, Market aplikasi Android serta dukungan yang sangat tinggi dari komunitas open source di dunia. Sehingga Android terus berkembang pesat baik dari segi jumlah device yang ada di dunia[11].

2.4.2 Perkembangan Android

1. Android versi 1.1

Android memang diluncurkan pertama kali pada tahun 2007, namun sistem operasi ini mulai dirilis dan diterapkan ke berbagai gadget pada tanggal 9 Maret 2009 silam. Android versi 1.1 merupakan Android awal yang dimana versi ini baru memberikan sentuhan dibebberapa aplikasinya seperti sistem antar muka bagi pengguna (user interface) yang lebih baik,serta beberapa aplikasi yang lain.

2. Android versi 1.5 (*Cupcake*)

Pada bulan Mei 2009 Android kembali mengalami perubahan versi. Android versi 1.1 kemudian disempurnakan dengan Android versi 1.5 atau yang dikenal sebagai Android *Cupcake*.

3. Android versi 1.6 (*Donut*)

Donut (versi 1.6) diluncurkan dalam tempo kurang dari 4 bulan semenjak peluncuran perdana Android *Cupcake*, yaitu pada bulan September 2009.

4. Android versi 2.0/2.1 (*Eclair*)

Masih ditahun yang sama, Android kembali merilis operating sistem versi terbarunya, yaitu Android versi 2.0/2.1 *Eclair*. Android *Eclair* diluncurkan oleh Google 3 bulan setelah peluncuran.

5. Android versi 2.2 (*Froyo: Frozen Yoghurt*)

Butuh 5 bulan bagi Google untuk melakukan regenerasi dari Android *Eclair* versi sebelumnya ke versi *Froyo Frozen Yoghurt*. Pada tanggal 20 Mei 2010, Android versi 2.2 alias Android *Froyo* ini dirilis.

6. Android versi 2.3 (*Gingerbread*)

7 bulan kemudian Android kembali melakukan gebrakan dengan merilis kembali Android versi 2.3 atau yangdikenal sebagai Android *Gingerbread*.

7. Android versi 3.0/3.1 (*Honeycomb*)

Pada bulan Mei 2011 Android versi 3.0/3.1 atau Android *Honeycomb* dirilis. Android *Honeycomb* merupakan sebuah sistem operasi Android yang tujuannya memang dikhususkan bagi penggunaan tablet berbasis Android.

8. Android versi 4.0 (*ICS: Ice Cream Sandwich*)

Android *ICS* atau *Ice Cream Sandwich* juga dirilis pada tahun yang sama dengan *Honeycomb*, yaitu pada bulan Oktober 2011.

9. Android versi 4.1 (*Jelly Bean*)

Android *Jelly Bean* merupakan versi Android yang terbaru pada saat ini. Salah satu gadget yang menggunakan sistem operasi *Jelly Bean* adalah Google Nexus 7 yang diprakarsai oleh ASUS, vendor asal Taiwan yang juga menjadi teman satu kampung halaman dengan Acer.

10. Android versi 4.4 (*Kit Kat*)

Kehadiran android *kitkat* merupakan peluncuran produk OS anyar yang diluncurkan pada 4 september 2013, sebelumnya banyak kabar beredar jikalau android akan meluncurkan OS baru yang bernama Android Key Lime Pie namun setelah di analisa tidak sesuai dengan ejaan orang umum, sehingga namanya diganti dengan OS Android *KitKat* yang sebagian besar orang sudah familiar dengan itu.

11. Android versi 5.0.2 (*Lollipop*)

Android *Lollipop* merupakan keberadaan OS Android yang memang saat ini sudah menjadi trend baru di industri smartphone, hal ini tak lepas dari keunikan dan kelebihan yang banyak di miliki dari OS tersebut. Kehadiran android versi ini amat di nanti oleh sekian banyak orang karna diharapkan sistem operasi *Lollipop* ini bias lebih baik dibandingkan versi-versi sebelumnya.

12. Android versi 6.0 (*Marshmallow*)

Android 6.0 *Marshmallow* adalah versi dari sistem operasi mobile Android. Pertama kali diperkenalkan Mei 2015 di Google I / O dibawah kode nama Android M, secara resmi dirilis pada Oktober 2015. Android *Marshmallow* memperkenalkan model izin aplikasi didesain ulang sekarang ada hanya delapan kategori izin, dan aplikasi yang tidak lagi secara otomatis diberikan semua hak akses mereka ditentukan pada waktu instalasi[12].

13. Android versi 7.0 (*Nougat*)

Nougat (versi 7.0) keluaran 2016 adalah salah satu upgrade terbesar untuk sistem operasi Android. Google menambahkan dukungan *multi-window* sehingga pengguna bisa menjalankan dua aplikasi secara bersamaan dalam jendela terpisah yang diposisikan secara berdampingan.

14. Android versi 8.0 (*Oreo*)

Setelah *Nougat*, Android *Oreo* (versi 8.0) yang dirilis pada 2017 menambah lebih banyak fitur *multi-tasking* seperti *Picture-in-Picture*. Bagian notifikasi ikut dirombak. Pengguna bisa mengatur notifikasi mana saja yang akan ditampilkan dan apa yang dilakukan perangkat saat menyuguhkannya. Selain OS Android versi “*reguler*”, Google juga membuat Android versi “ringan” bernama Android Go Edition yang diambil dari basis Android *Oreo*. OS ini khusus ditujukan supaya bisa dijalankan dengan lancar oleh perangkat-perangkat entry level Android Go.

15. Android versi 9.0 (*Pie*)

Inilah Android teranyar yang baru saja diresmikan pada Agustus lalu. Android *Pie* (versi 9.0) mengganti tiga tombol navigasi dengan tombol tunggal berbentuk elips. Karenanya, Android *Pie* pun lebih mengandalkan gestur gerakan-gerakan jari untuk melakukan hal-hal seperti *multitasking*. Fitur lain termasuk “*Digital Wellbeing*” untuk menginformasikan soal pola pemakaian perangkat, *Adaptive Battery* untuk membatasi pemakaian baterai oleh aplikasi, dan *App Action* yang langsung menjalankan fitur aplikasi dari *app drawer*.

2.4.3 Android SDK (*Software Development Kit*)

Android SDK mencakup perangkat tools pengembangan yang komprehensif. Android SDK terdiri dari debugger, libraries, handset emulator, dokumentasi, contoh kode program dan tutorial. Saat ini Android sudah mendukung arsitektur x86 pada Linux (distribusi Linux apapun untuk desktop modern), Mac OS X 10.4.8 atau lebih, Windows XP atau Vista. Persyaratan mencakup JDK, Apache Ant dan Python 2.2 atau lebih. IDE yang didukung secara resmi adalah Eclipse 3.2 atau lebih dengan menggunakan plugin *Android Development Tools* (ADT), dengan ini pengembang dapat menggunakan IDE untuk mengedit dokumen *Java* dan *XML* serta menggunakan peralatan command line untuk menciptakan, membangun, melakukan debug aplikasi Android dan pengendalian perangkat Android (misalnya *reboot*, menginstal paket perangkat lunak).

2.4.4 *Android Development Tools* (ADT)

Android Development Tools adalah penghubung antara IDE Eclipse dengan Android SDK. *Eclipse* dapat digunakan untuk mengembangkan aplikasi android dengan menggunakan *Android Development Tools* yang mempunyai fungsi:

- a. Dapat membuat, menguji dan menyusun aplikasi Android yang berjalan di *smartphone* Android.

- b. Mensimulasikan seluruh pengalaman pengguna online dan offline untuk berbagai jenis *smartphone* Android dari desktop.
- c. Memungkinkan untuk memanfaatkan keunggulan pengembangan unik untuk platform Android[13].

2.5 *Eclipse*

Eclipse adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (*platform independent*). Berikut ini adalah sifat dari *Eclipse* :

- 1) *Multi-platform*: Target sistem operasi *Eclipse* adalah Microsoft Windows, Linux, Solaris, AIX, HP-UX dan Mac OS X.
- 2) *Multilanguage*: *Eclipse* dikembangkan dengan bahasa pemrograman *Java*, akan tetapi *Eclipse* mendukung pengembangan aplikasi berbasis bahasa pemrograman lain seperti C/C++, Cobol, Python, Perl, PHP, dan lain sebagainya.
- 3) *Multi-role*: Selain sebagai IDE untuk pengembangan aplikasi. *Eclipse* bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak seperti dokumentasi, pengujian perangkat lunak, pengembangan web, dan lain sebagainya.

Eclipse merupakan salah satu IDE favorit karena gratis dan *open source*. *Open source* berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini, Selain itu kelebihan dari *Eclipse* yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan membuat komponen yang disebut *plugin*[14].

Gambar 2.2. Eclipse

2.6 *Java Development Kit (JDK)*

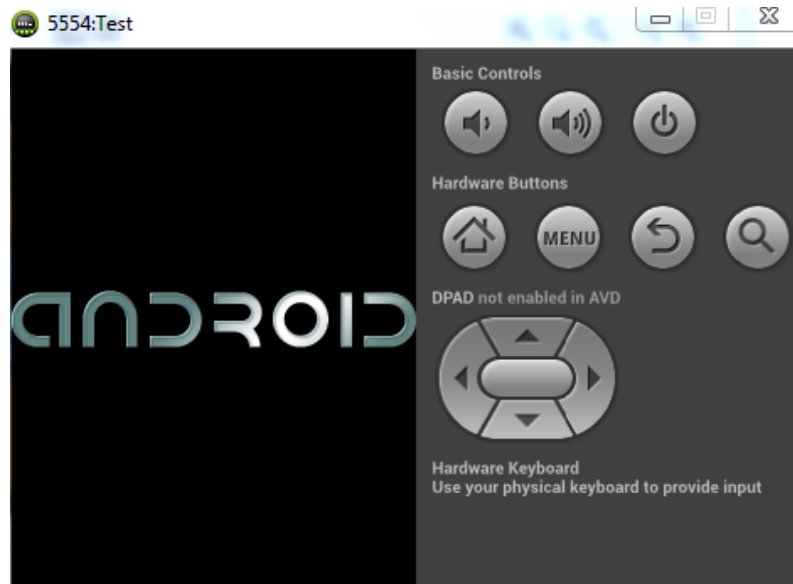
Java Development Kit (JDK) adalah sekumpulan perangkat lunak yang dapat kamu gunakan untuk mengembangkan perangkat lunak yang berbasis Java, sedangkan JRE adalah sebuah implementasi dari *Java Virtual Machine* yang benar-benar digunakan untuk menjalankan program *java*. Biasanya, setiap JDK berisi satu atau lebih JRE dan berbagai alat pengembangan lain seperti sumber *compiler java, bundling, debuggers, development libraries* dan lain



sebagainya[12].

2.7 *Android Virtual Device (AVD)*

Android Virtual Device merupakan emulator untuk menjalankan aplikasi android (<http://developer.android.com/guide/developing/devices/index.html>), yang tampilannya dapat dilihat pada gambar 1. Setiap AVD terdiri dari sebuah profil perangkat keras yang dapat mengatur pilihan untuk menentukan fitur dari *hardware emulator*. Misalnya, menentukan apakah menggunakan perangkat kamera, apakah menggunakan keyboard QWERTY fisik atau tidak, berapa banyak memori internal, dan lain-lain. AVD juga memiliki sebuah pemetaan versi Android, maksudnya kita menentukan versi dari platform Android akan berjalan pada *emulator* [15].



Gambar 2.3. Tampilan AVD

2.8 XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program *Apache HTTP Server*, *MySQL database*, dan penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl. Nama *XAMPP* merupakan singkatan dari X (empat sistem operasi apapun), *Apache*, *MySQL*, *PHP* dan *Perl*. Program ini tersedia dalam *GNU General Public License* dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis. Untuk mendapatkannya dapat mendownload langsung dari web resminya[16].



Gambar 2.4. XAMPP Logo

2.9 PHP (*Hypertext Preprocessor*)

PHP atau kependekan dari *Hypertext Preprocessor* adalah salah satu bahasa pemrograman *open source* yang sangat cocok atau dikhususkan untuk pengembangan web dan dapat ditanamkan pada sebuah skripsi *HTML*. Bahasa *PHP* dapat dikatakan menggambarkan beberapa bahasa pemrograman seperti *C*, *Java*, dan *Perl* serta mudah untuk dipelajari.

PHP merupakan bahasa *scripting server – side*, dimana pemrosesan datanya dilakukan pada sisi server. Sederhananya, serverlah yang akan menerjemahkan skrip program, baru kemudian hasilnya akan dikirim kepada client yang melakukan permintaan.

Adapun pengertian lain *PHP* adalah akronim dari *Hypertext Preprocessor*, yaitu suatu bahasa pemrograman berbasis kode – kode (*script*) yang digunakan untuk mengolah suatu data dan mengirimkannya kembali ke web browser menjadi kode *HTML*”.

Menurut Kustiyarningsih (2011:114), “*PHP* (atau resminya *PHP: Hypertext Preprocessor*) adalah skrip bersifat *server – side* yang ditambahkan ke dalam *HTML*”.

Pada prinsipnya server akan bekerja apabila ada permintaan dari client. Dalam hal ini client menggunakan kode-kode *PHP* untuk mengirimkan permintaan ke server.

Sistem kerja dari *PHP* diawali dengan permintaan yang berasal dari halaman website oleh browser. Berdasarkan URL atau alamat website dalam jaringan internet, browser akan menemukan sebuah alamat dari webserver, mengidentifikasi halaman yang dikehendaki, dan menyampaikan segala informasi yang dibutuhkan oleh webserver.

Selanjutnya webserver akan mencari berkas yang diminta dan menampilkan isinya di browser. Browser yang mendapatkan isinya segera menerjemahkan kode *HTML* dan menampilkannya. Lalu bagaimana apabila yang dipanggil oleh user adalah halaman yang mengandung script *PHP*? . Pada prinsipnya sama dengan memanggil kode *HTML*, namun pada saat permintaan dikirim ke web-server, web-server akan memeriksa tipe file yang diminta user. Jika tipe file yang diminta adalah *PHP*, maka akan memeriksa isi script dari halaman *PHP* tersebut.

Apabila dalam file tersebut tidak mengandung script *PHP*, permintaan user akan langsung ditampilkan ke browser, namun jika dalam file tersebut mengandung script *PHP*, maka proses akan dilanjutkan ke modul *PHP* sebagai mesin yang menerjemahkan *script-script PHP* dan mengolah script tersebut, sehingga dapat dikonversikan ke *kode-kode HTML* lalu ditampilkan ke browser user.

2.10 MySQL

Menurut Kustiyarningsih (2011:145), “*MySQL* adalah sebuah basis data yang mengandung satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau sejumlah field..

Menurut Wahana Komputer (2010:21), MySQL adalah database server open source yang cukup populer keberadaannya. Dengan berbagai keunggulan yang dimiliki, membuat software database ini banyak digunakan oleh praktisi untuk membangun suatu project. Adanya fasilitas *API (Application Programming Interface)* yang dimiliki oleh *MySQL*, memungkinkan bermacam – macam aplikasi komputer yang ditulis dengan berbagai Bahasa pemrograman dapat mengakses basis data *MySQL*.

Tipe data *MySQL*, menurut Kustiyahningsih (2011:147), “Tipe data *MySQL* adalah data yang terdapat dalam sebuah table berupa field – field yang berisi nilai dari data tersebut. Nilai data dalam field memiliki tipe sendiri – sendiri”[17].

2.11 Unified modelling language (UML)

Pengertian UML menurut Nugroho (2010) ”*UML (unified modeling language)* adalah Bahasa pemodelan untuk sistem atau perangkat lunak yang berparadigma berorientasi pada objek”. Pemodelan (*modeling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang kompleks sedemikian rupasehingga lebih mudah dipelajari dan dipahami.

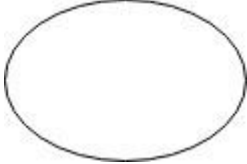



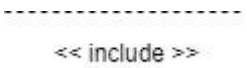
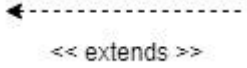
Adapun tujuan pemodelan (dalam rangka pengembangan sistem/perangkat lunak aplikasi) yaitu sebagai sarana analisis, pemahaman visualisasi, dan komunikasi antar anggota tim pengembang (saat seorang analisis atau perancang perangkat lunak bekerja dalm tim yang beranggotakan beberapa atau banyak anggota), serat sebagai sarana dokumentasi(yang bermanfaat untuk menelaah perilaku perangkat lunak secara seksama serta bermanfaat untuk melakukan pengujian terhadap perangkat lunak yang telah selesai digunakan.

Para pakar di bidang perancangan perangkat lunak pada sekitar tahun 1980-1990 mulai bekerja dengan bahasa pemrograman berorientasi objek (*OOP - Object Oriented Programming*). Dengan demikian, diperlukan metodologi yang lebih sesuai dalam hal ini adalah UML yang merupakan metodologi kolaborasi antara metode-metode Booch yang dikembangkan oleh Graddy Booch, *OMT (Object Modeling Technique)* yang dikembangkan oleh DR. James Rumbaugh, serta *OOSE (Object Oriented Software Engineering)* yang dikembangkan oleh Ivar Jacobson, dan beberapa metode lainnya. Merupakan metodologi yang paling sering atau paling tepat digunakan saat ini untuk mengadaptasi penggunaan bahasa-bahasa pemrograman yang berparadigma pada pemrograman berbasis objek.[4]

2.11.1 Use Case Diagram

Munawar (2005 : 64) menyatakan bahwa *use case* adalah deskripsi fungsi dari sebuah sistem dari perspektif pengguna. *Use case* bekerja dengan cara deskripsikan tipikal interaksi pada antar user (pengguna) sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai[12].

Tabel 2.1 *Use Case Diagram*[18].




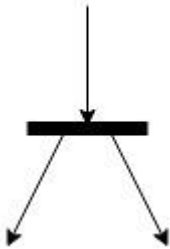
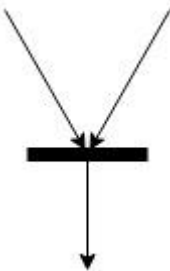
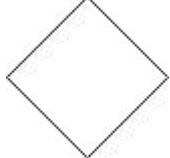
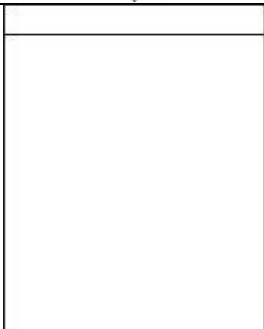
Gambar	Keterangan
	<i>Use Case</i> menggambarkan fungsional yang disediakan sistem sebagai unit-unit yang bertukar oesan antar aktir, yang dinyatakan dengan menggunakan kata kerja.
	<i>Actor</i> atau aktor adalah <i>abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasi aktir, ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem. Orang tau sistem bisa muncul dalam beberapa peran. Perlu dicatat bahwa aktor berinteraksi dengan <i>use case</i> , tetapi tidak memiliki control terhadap <i>use case</i> .
Gambar	Keterangan
	Asosiasi antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan data.
	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.
	<i>Include</i> , merupakan di dalam <i>use case</i> lain (<i>required</i>) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.
	<i>Extend</i> , merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

Sumber : A. Hendini, "Pemodelan UML sistem informasi Monitoring Penjualan dan stok barang," *Pemodelan Uml Sist. Inf. Monit. Penjualan Dan Stok Barang (Studi Kasus Distro Zhezha Pontianak)*, vol. IV, no. 2, pp. 107–116, 2016.

2.11.2 Activity Diagram

Activity diagram menurut Munawar (2005: 109) adalah teknik untuk mendeskripsikan logika prosedural, proses bisnis dan aliran kerja dalam banyak kasus. *Activity* diagram mempunyai peran seperti halnya *flowchart*, akan tetapi perbedaannya dengan *flowchart* adalah *activity* diagram bisa mendukung perilaku paralel sedangkan *flowchart* tidak bisa[12].

Tabel 2.2 *Activity* Diagram[18].


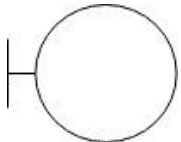


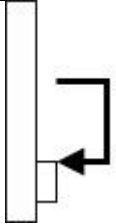


Gambar	Keterangan
	<i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktivitas.
	<i>End Point</i> , akhir aktivitas.
	<i>Activities</i> , menggambarkan suatu proses atau kegiatan bisnis.
	<i>Fork</i> /percabangan, digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	<i>Join</i> (penggabungan) atau <i>rake</i> , digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Point</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> atau <i>false</i> .
	<i>Swimlane</i> , pembagian <i>activity diagram</i> untuk menunjukkan siapa melakukan apa.

Sumber : A. Hendini, "Pemodelan UML sistem informasi Monitoring Penjualan dan stok barang," *Pemodelan Uml Sist. Inf. Monit. Penjualan Dan Stok Barang (Studi Kasus Distro Zhezha Pontianak)*, vol. IV, no. 2, pp. 107–116, 2016.

2.11.3 Sequence Diagram

Diagram sequence merupakan salah satu yang menjelaskan bagaimana suatu operasi itu dilakukan message (pesan) apa yang dikirim dan kapan pelaksanaannya[12]

.Tabel 2.3 *Sequence Diagram*[18].



Gambar	Keterangan
	<i>Entity Class</i> , merupakan bagian dari sistem yang berisi sekumpulan kelas berupa entitas-entitas yang membentuk gambaran awal sistem dan menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi <i>interface</i> atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan <i>form entry</i> dan <i>form cetak</i> .
	<i>Control Class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas. Contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek.
	<i>Message</i> , simbol mengirim pesan antar <i>class</i> .
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.
	<i>Activation</i> , mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivitas sebuah operasi.
	<i>Lifeline</i> , garis titik-titik yang berhubungan dengan objek sepanjang <i>lifeline</i> terdapat <i>activation</i> .

Sumber : A. Hendini, "Pemodelan UML sistem informasi Monitoring Penjualan dan stok barang," *Pemodelan Uml Sist. Inf. Monit. Penjualan Dan Stok Barang (Studi Kasus Distro Zhezha Pontianak)*, vol. IV, no. 2, pp. 107–116, 2016.

2.11.4 Component Diagram

Component diagram merupakan diagram yang memiliki tujuan khusus untuk difokuskan pada perangkat lunak dan komponen perangkat keras. Diagram komponen digunakan untuk menggambarkan komponen. Diagram komponen (*component diagram*) digunakan untuk memodelkan aspek fisik dari suatu sistem, komponen diagram digunakan untuk memvisualkan organisasi dan hubungan antara komponen dalam satu sistem. Diagram ini juga digunakan untuk menjelaskan cara sistem dapat dieksekusi[19].

Tabel 2.4 *Component Diagram*[18].

Gambar	Keterangan
	Pada <i>deployment diagram</i> , komponen-komponen yang ada diletakkan didalam kode untuk memastikan keberadaan posisi mereka.
	Sebuah <i>association</i> digambarkan sebagai sebuah garis yang menghubungkan dua node yang mengindikasikan jalur komunikasi antara <i>element-element hardware</i> .

Sumber : A. Hendini, "Pemodelan UML sistem informasi Monitoring Penjualan dan stok barang," *Pemodelan Uml Sist. Inf. Monit. Penjualan Dan Stok Barang (Studi Kasus Distro Zhezha Pontianak)*, vol. IV, no. 2, pp. 107–116, 2016.

2.12 AMIK LabuhanBatu

Akademi Manajemen Informatika Komputer (AMIK) Labuhanbatu merupakan salah satu Perguruan Tinggi yang tertua dan hanya satu-satunya di Kabupaten Labuhanbatu untuk Program Diploma III yang sudah terakreditasi "B" dengan program studi Manajemen Infomatika Komputer. Berdiri sejak tahun 2001 berdasarkan keputusan Menteri Pendidikan dan Kebudayaan Republik Indonesia No.231/D/O/2001 Tentang pemberian status terdaftar kepada program studi Diploma III pada AMIK Labuhanbatu berkedudukan di Kabupaten Labuhanbatu Sumatera Utara.

AMIK Labuhanbatu berada di bawah Yayasan Universitas Labuhanbatu, bersama 4 Sekolah Tinggi lainnya yaitu Sekolah Tinggi Keguruan Ilmu Pendidikan (STKIP) Labuhanbatu, Sekolah Tinggi Ilmu Ekonomi (STIE) Labuhanbatu, Sekolah Tinggi Ilmu Hukum (STIH) Labuhanbatu, dan Sekolah Tinggi Ilmu Pertanian (STIPER) Labuhanbatu. Yayasan Universitas Labuhanbatu (YULB) di didirikan oleh DR.H.Amarullah Nasution, SE.,MBA pada tahun 1998 yang di kenal sebagai tokoh pendidikan di Sumatera Utara.

Dalam sejarahnya, AMIK Labuhanbatu dipimpin oleh beberapa orang direktur,yaitu:

1. Tahun 2000 – 2005 dipimpin oleh Syafril Rambe,ST.,MM
2. Tahun 2005 – 2010 dipimpin oleh Awaluddin Hasibuan,ST.,M.Kom
3. Tahun 2010 – 2013 dipimpin oleh Selamat Subagio,S.Kom.,M.kom
4. Tahun 2013 – 2019 dipimpin oleh Sudi Suryadi,S.kom..M.kom

5. Tahun 2019 sampai sekarang dipimpin oleh Deci Irmayani, S.kom., M.kom

Program Studi (Prodi) Manajemen Informatika (MI) didirikan pada tahun 2001 berdasarkan Surat Izin Penyelenggaraan Program Studi dan Pendirian AMIK Labuhan Batu dengan No. 231/D/O/2001. Prodi Manajemen Informatika merupakan pendidikan tinggi tingkat Diploma III dengan masa studi paling cepat 6 semester (3 tahun) dan paling lambat 10 semester (5 tahun).

Prodi Manajemen Informatika berfokus kepada penekanan terhadap pentingnya nilai strategis suatu informasi sebagai sumber daya dalam suatu organisasi dalam pencapaian tujuan organisasi tersebut. Prodi ini mempelajari konsep dan teori penerapan sistem informasi sebuah organisasi, terutama dalam proses penciptaan, pengolahan, penyimpanan, pendistribusian, dan pengawasan informasi dalam suatu organisasi.

Prodi Manajemen Informatika akan membekali lulusannya dalam kemampuan dibidang pengelolaan sistem informasi yang mengacu kepada perkembangan teknologi informasi.

2.12.1 Visi AMIK LabuhanBatu

Sebagai Perguruan Tinggi Yang Menjadi Salah Satu Pusat Pengembangan Sumber Daya Manusia, Ilmu Pengetahuan, dan Riset di Bidang Informatika Komputer Pada Tahun 2027 di Sumatera Utara.

2.12.2 Misi AMIK LabuhanBatu

1. Menyelenggarakan pendidikan perguruan tinggi yang berkualitas, handal, dan kompetitif serta unggul ditingkat regional dan bersaing ditingkat nasional.
2. Melaksanakan kegiatan penelitian yang berkaitan dengan pengembangan keilmuan dibidang informatika.
3. Menyelenggarakan kegiatan pengabdian kepada masyarakat sekitar secara berkesinambungan yang berorientasi kepada peningkatan sumber pemberdayaan masyarakat dibidang informatika.
4. Meningkatkan hubungan kerjasama yang strategis dengan lembaga pemerintah dan atau non pemerintah baik ditingkat lokal, nasional, atau internasional.
5. Menyelenggarakan Perguruan Tinggi yang sehat dan bermanfaat bagi masyarakat.

2.12.3 Tujuan AMIK LabuhanBatu

- Menghasilkan lulusan profesional di bidang informatika yang memiliki kompetensi, kecerdasan intelektual, dan berjiwa enterprenership yang unggul ditingkat regional serta mampu bersaing di tingkat nasional.
- Menghasilkan penelitian-penelitian dosen dan mahasiswa yang berkaitan dengan pengembangan keilmuan informatika

- Menghasilkan berbagai karya dalam kegiatan pengabdian kepada masyarakat dengan tujuan meningkatkan pemberdayaan masyarakat di bidang informatika
- Menjalin kerja sama yang berkesinambungan dengan lembaga pemerintahan dan non pemerintah, alumni, serta stakeholder untuk meningkatkan kompetensi lulusan yang berorientasi ke masa depan menjadi perguruan tinggi yang sehat dan bermanfaat bagi masyarakat



Gambar 2.5 AMIK Labuhan Batu

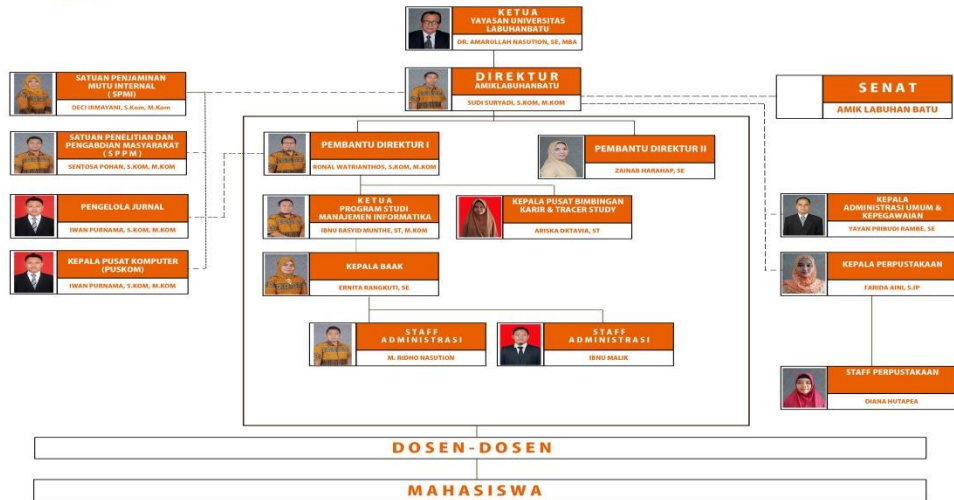
2.12.4 Struktur Organisasi AMIK LabuhanBatu

Struktur organisasi adalah salah satu susunan dan hubungan antara tiap bagian serta posisi yang ada pada suatu organisasi atau perusahaan dalam menjalankan suatu kegiatan operasional untuk mencapai tujuan yang diharapkan dan di inginkan.

Ketua Yayasan	: DR.H.Amarullah Nasution, SE.,MBA
Direktur	: Decy Irmayani, S.Kom.,M.Kom
Direktur II Bid. Keuangan	: Zainab Harahap, SE.M.Si
Ketua Prodi MI	: Marnis Nasution, S.Kom.,M.Kom
Ketua SPMI	: Decy Irmayani, S.Kom.,M.Kom



STRUKTUR ORGANISASI INSTITUSI AMIK LABUHAN BATU



Keterangan :
 Garis Instruksi ———
 Garis Koordinasi - - - - -

- Ketua LPPM : Sentosa Pohan ,S.Kom.,M.Kom
- Kepala BAAK : M. Ridho Nasution, SH
- Staff Administ : Ibnu Malik,S.P
- Kepala Perpustakaan : Farida Aini, S.IP
- Kepala Lab. Komputer : Iwan Purnama, S.Kom.,M.Kom

Gambar 2.6. Struktur Organisasi Institusi AMIK Labuhanbatu

2.12.5 Tugas Pokok dan Fungsi Struktur AMIK LabuhanBatu

Tugas Pokok dan Fungsi masing-masing Struktur AMIK Labuhan Batu Adalah Sebagai berikut :

1) Ketua Yayasan

a. Tugas Pokok

- Melaksanakan fungsi manajemen tertinggi di universitas.
- Memimpin penyelenggaraan Tri Dharma Perguruan Tinggi.
- Memimpin senat universitas.
- Memimpin Evaluasi diri, Validasi dan Peningkatan Akreditasi Program Studi, penerapan penjaminan mutu
- Menjalin hubungan dan lobbying antar lembaga
- Menandatangani dokumen-dokumen resmi ditingkat universitas yang telah mendapat persetujuan Senat dan Badan Penyelenggara.
- Menghadiri dan mewakili Universitas di forum-forum resmi.

b. Fungsi

- Menilai, mempromosikan, memberhentikan pejabat dalam jajarannya sesuai prosedur yang berlaku.
- Menyiapkan pegawai untuk dicalonkan dalam pengangkatan, upgrading, pengembangan dsb sesuai prosedur yang berlaku.
- Bertindak untuk dan atas nama Yayasan dalam melakukan lobby kerjasama antar lembaga sesuai prosedur yang berlaku.
- Menjabarkan job description bawahannya sesuai kebutuhan.
- Mendelegasikan semua tugas-tugas kepada bawahan tanpa mengalihkan tanggungjawab.

2) Direktur

a. Tugas Pokok

- Memegang wewenang tertinggi dan memikul tanggung jawab mengenai seluruh kegiatan sekolah Tinggi, baik bersifat strategis, teknis maupun administratif.
- Melaksanakan fungsi pimpinan yang meliputi perencanaan, pembuatan keputusan, melakukan semua koordinasi, pengarahan, pengendalian dan pengawasan, serta penyempurnaan pelaksanaan tugas bagi tercapainya seluruh tujuan sekolah tinggi.
- Bertanggungjawab terhadap pelaksanaan jaminan mutu dalam rangka meningkatkan daya saing institusi Menyelenggarakan kegiatan public relation dengan stakeholder.
- Mewakili sekolah Tinggi dalam berbagai forum , baik di dalam maupun di luar negeri.
- Bertanggungjawab terhadap kewajiban dan hak sekolah tinggi terhadap yayasan.
- Menyelenggarakan komunikasi konstruktif bersama dengan SENAT AKADEMIK Sekolah Tinggi.
- Melaksanakan koordinasi dnegan Senat Akademik dalam penyusunan renstra 5 tahunan Akademi.
- Menyusun Rencana Anggaran dan Belanja tahunan kepada yayasan.

- Mengelola, memonitor dan mengevaluasi kinerja bawahan.
- Menyusun laporan-laporan pertanggungjawaban kepada yayasan
- Menyampaikan laporan-laporan kepada pemerintah
- Melaksanakan tugas-tugas lainnya yang berhubungan dengan proses pengembangan Sekolah Tinggi.

b. Fungsi

- Menyusun dan menetapkan kebijakan baik yang bersifat strategis, teknis dan administratif.
- Mengawasi dan mengevaluasi pelaksanaan RKAT dan kinerja pembantu ketua, Kepala Humas Admisi, Sijar, PJM.
- Berkoordinasi dengan pihak Yayasan dan Senat Akademik.
- Mengambil langkah langkah strategis dan penyelesaian masalah di lingkungan akademi

3) Wakil Direktur I Bid.Akademi

a. Tugas Pokok

- Membantu menyusun Rencana Strategis, Rencana Kerja dan Anggaran Tahun Sekolah Tinggi di bidang akademik, penelitian dan pengabdian masyarakat.
- Menyusun perencanaan, penyelenggaraan, pengembangan dan melakukan evaluasi kebijakan, terhadap kegiatan beserta anggaran bidang akademik, penelitian dan pengabdian masyarakat berbasis SPMI.

- Menyusun perencanaan, menyelenggarakan, mengembangkan dan melakukan evaluasi kebijakan di bidang sumber daya manusia yang berkaitan dengan pengembangan kualitas TRI DARMA PT.
- Menjamin terlaksananya siklus pengembangan materi pembelajaran
- Membantu dalam merencanakan, mengembangkan dan optimalisasi penggunaan sarana prasarana yang berkaitan dengan kualitas TRI DARMA PT.
- Merencanakan dan mengembangkan program studi baru untuk berbagai tingkat/ bidang
- Merencanakan dan mengembangkan kapabilitas program studi.
- Memberikan dukungan pelaksanaan kerja sama TRI DARMA PT .
- Mengkoordinasi perencanaan, pelaksanaan, monitoring, evaluasi dan koreksi pelaporan epsbed dari masing-masing prodi .
- Merencanakan dan melakukan koordinasi dalam proses akreditasi/ijin penyelenggaraan jurusan.
- Mengelola, memonitor dan mengevaluasi kinerja bawahan .
- Menyusun laporan kegiatan dan keuangan di bidang Tri Dharma secara berkala kepada ketua akademi.
- Melaksanakan tugas-tugas lain untuk pekerjaan yang diperintahkan atasan yang mendukung tujuan jabatan

b. Fungsi

- Mengevaluasi dan melakukan koreksi kebijakan dan kegiatan beserta anggaran TRI DARMA PT berbasis SPMI.
- Mengevaluasi SDM yang berkaitan dengan kualitas TRI DARMA PT.
- Mengevaluasi pelaporan epsbed dari masing-masing prodi .

4) Wakil Direktur II Bid.Keuangan

a. Tugas Pokok

- Melakukan pelayanan di bidang keuangan kepada mahasiswa, dosen dan staf.
- Melakukan pembukuan sesuai standar akuntansi
- Melakukan validasi data.

b. Fungsi

- Mengambil kebijakan jika atasan tidak berada di tempat.

5) Wakil Direktur III. Bidang Mahasiswa

a. Tugas Pokok

- Menyusun rencana program kerja Politeknik di bidang kemahasiswaan sebagai pedoman pelaksanaan tugas.
- Memberi tugas dan juga pengarahan kepada pimpinan unit kerja bidang kemahasiswaan untuk kelancaran pelaksanaan tugas.
- Mengkoordinasikan pimpinan unit kerja di bidang kemahasiswaan agar terjalin kerjasama yang baik.

- Menyelidiki pelaksanaan tugas pimpinan unit kerja di bidang kemahasiswaan agar pelaksanaannya sesuai dengan ketentuan yang berlaku.
- Mengevaluasi pelaksanaan tugas pimpinan unit kerja di bidang kemahasiswaan.
- Menyusun petunjuk teknis dan menyusun saran alternatif di bidang kemahasiswaan sebagai bahan masukan pimpinan Politeknik.
- Menyusun konsep pengembangan pelayanan kesejahteraan mahasiswa.
- Mengkoordinasikan pelaksanaan kegiatan kemahasiswaan di lingkungan Politeknik.
- Mengevaluasi hasil pelaksanaan kegiatan kemahasiswaan di lingkungan Politeknik.
- Menyusun laporan di bidang kemahasiswaan.
- Melaksanakan tugas lain yang diberikan oleh atasan.

b. Fungsi

- Pelaksanaan pembinaan mahasiswa oleh seluruh dosen dalam pengembangan sikap dan orientasi serta kegiatan mahasiswa antara lain dalam seni budaya dan olahraga serta bakti sosial sebagai bagian pembinaan sivitas akademika yang merupakan sebagian dari tugas pendidikan tinggi pada umumnya.
- Pelaksanaan usaha kesejahteraan mahasiswa serta usaha bimbingan dan penyuluhan bagi mahasiswa.

- Pelaksanaan usaha pengembangan daya penalaran mahasiswa yang sudah diprogramkan oleh Pembantu Direktur I.
- Kerjasama dengan semua pihak dalam setiap usahadi bidang kemahasiswaan, pengabdian kepada masyarakat dan usaha penunjangnya.
- Terciptanya iklim pendidikan yang baik dalam kampus dan membantu pelaksanaan program pembinaan pemeliharaan kesatuan dan persatuan bangsa berdasarkan Pancasila dan Undang-Undang Dasar 1945.
- Pelaksanaan kegiatan kemahasiswaan dalam rangka usaha pembangunan yang tetap dilandasi nilai-nilai dan tanggungjawab yang bersifat akademik.

6) Ketua Prodi Manajemen Informatika

a. Tugas Pokok

- Menyusun rencana, memberi petunjuk dan mengevaluasi pelaksanaan kegiatan pendidikan dan pengajaran yang dilaksanakan dosen di lingkungan program studi berdasarkan ketentuan yang berlaku untuk kelancaran pelaksanaan tugas.

b. Fungsi

- Bertanggung jawab atas keseluruhan pelaksanaan kegiatan di Program Studi.
- Mengusulkan staf yang akan duduk dalam jabatan struktural sekretaris prodi, kepala laboratorium dan koordinator laboratorium dll, sesuai kebut program studi.

- Mengusulkan staf untuk melaksanakan kegiatan pendidikan, pelatihan, produksi, konsultasi dan rekayasa.
- Mengusulkan staf yang kompeten untuk melaksanakan kegiatan yang berkaitan dengan pengembangan program studi.
- Bertanggungjawab langsung bila pejabat struktural lainnya berhalangan hadir, sesuai dengan lingkup kerjanya.

7) Ketua SPM

a. Tugas Pokok

- Melakukan monitoring dan evaluasi akan jalannya kegiatan di lingkungan universitas.
- Membuat perangkat evaluasi untuk setiap bagian yang ada dilingkungan universitas.

b. Fungsi

- Menjamin mutu gelar akademik yang diberikan sesuai dengan visi, misi dan tujuan yang ada.
- Menjamin bahwa setiap kegiatan sesuai dengan Standar Operasional Prosedur yang ada.

8) KETUA LPPM

a. Tugas Pokok

- Memimpin penyelenggaraan Tri Dharma Perguruan Tinggi yaitu Dharma penelitian dan Dharma pengabdian masyarakat di tingkat Universitas.

- Menyiapkan sarana penelitian dan pengabdian masyarakat di lingkungan Universitas.
- Menandatangani dokumen-dokumen resmi dilingkungan lembaga penelitian dan pengabdian masyarakat.
- Merancang dan membangun model kerjasama dalam bidang penelitian dan pengabdian kepada masyarakat baik dengan instansi pemerintah, swasta, perorangan maupun lembaga kemasyarakatan lainnya.
- Melibatkan segenap civitas akademika universitas dalam penelitian dan pengabdian masyarakat.

b. Fungsi

- Menyeleksi bahan penelitian yang akan diterbitkan di jurnal penelitian UN.
- Mengatur kegiatan KKN, KCU, Magang kerja .
- Menentukan dan menyeleksi usulan proposal penelitian di lingkungan universitas.

9) Kepala BAAK

a. Tugas Pokok

- Pelaksanaan administrasi pendidikan dan evaluasi.
- Pelaksanaan pemrosesan/penerbitan ijazah, transkrip dalam Bahasa Indonesia dan Bahasa Inggris, serta Surat Keterangan Pendamping

Ijasah (SKPI), dan atau akta bagi lulusan program studi kependidikan.

- Pelaksanaan registrasi mahasiswa baru dan lama.
- Pelaksanaan pembuatan statistic atau grafik.
- Pelaksanaan penerbitan kartu mahasiswa (KTM).
- Pelaksanaan administrasi sarana pendidikan.
- Pendataan ruang kuliah dan laboratorium.
- Pelaksanaan administrasi kerjasama.
- Pengurusan ijin ke luar negeri bagi dosen dan mahasiswa
- Pengurusan paspor dinas dan bebas fiskal.
- Koordinator pelaksanaan wisuda.

b. Fungsi

- Merupakan unsur pimpinan yang membantu Rektor dibidang administrasi akademik dan kemahasiswaan yang berada di bawah dan bertanggung jawab langsung kepada Rektor dan sehari-hari di bawah pembinaanya dilakukan oleh Wakil Rektor I Bidang Akademik, Wakil Rektor III Bidang Kemahasiswaan dan Wakil Rektor IV bidang informasi dan komunikasi dan bidang Kerjasama

10) Staff Administrasi BAAK

a. Tugas Pokok

- Membantu demi kelancaran semua pelaksanaan tugas yang berkaitan dengan kegiatan dilingkungan bagian administrasi akademik.

- Membantu dan melaksanakan serta melayani kegiatan yang berkaitan dengan pendaftaran mahasiswa baru, heregistrasi mahasiswa, pemrograman mata kuliah, pencetakkan absensi kuliah mahasiswa, pembagian KHS dan pengecekan nilai hasil ujian bila ada komplim dari mahasiswa.
- Mengarsipkan semua dokumen terkait dengan akademik : Buku laporan registrasi, buku laporan wisuda dan semua KHS dari berbagai Fakultas/Jurusan tiap-tiap semester.
- Menerima dan mendokumenkan serta mengarsipkan nilai matakuliah dosen pengampu mata kuliah dari semua fakultas/jurusan.
- Mengarsipkan dokumen berupa fotocopy nilai (transkrip alumni beserta ijazahnya) dan memberikan ijazah dan traskrip asli kepada alumni yang bersangkutan.
- Melaksanakan pengetikan surat-surat keterangan antara lain: keterangan masih studi, keterangan bea siswa, keterangan perpustakaan, keterangan lulus, keterangan cuti studi, keterangan mutasi studi dan pemrosesan kartu tanda mahasiswa (KTM).
- Pengetikan surat keputusan tentang susunan kepanitian kegiatan yang berkaitan dengan bagian administrasi akademik

b. Fungsi

- Bekerja sama dengan staf administrasi akademik dan staf unit kerja/bagian lain dilingkungan Y-ULB .

- Melaporkan hasil kegiatan dan bertanggung jawab kepada Kepala Sub Bagian Administrasi Pendidikan dan Pengajaran dan Kepala Bagian Administrasi Akademik.

11) Kepala Perpustakaan

a. Tugas Pokok & Fungsi

- Menyusun rencana strategi perpustakaan.
- Menyusun rencana dan program kerja tahunan.
- Menyusun program pengembangan Perpustakaan.
- Mengorganisasikan sumber daya dan dana.
- Membimbing dan membina sumber daya manusia di unitnya.
- Mengkoordinasikan penyelenggaraan bidang-bidang yang ada di unitnya.
- Melakukan monitoring & evaluasi pelaksanaan tugas layanan teknis, layanan pengguna, pengembangan dan kerjasama, serta administrasi dan kerumahtanggaan.
- Menyusun laporan berkala; bulanan, triwulan, semester dan tahunan kegiatan perpustakaan kepada Ketua Yayasan.

12) Kepala Lab. Komputer

a. Tugas Pokok

- Merencanakan kegiatan pendidikan, penelitian, dan pengabdian pada masyarakat di laboratorium.
- Menyusun rencana operasional dan pengembangan laboratorium.

- Memberikan pelayanan bagi civitas akademika untuk melakukan pengembangan ilmu pengetahuan dan teknologi.
- Menyiapkan jadwal kegiatan akademik yang dilaksanakan dalam laboratorium.
- Mengkoordinasikan segala kegiatan akademik yang dilaksanakan dalam laboratorium Melakukan pembinaan kepada anggota laboratorium.
- Menjalin kerjasama dengan pihak luar dalam rangka resource sharing dan pemberdayaan laboratorium.
- Melakukan pemantauan dan evaluasi atas ketersediaan sarana prasarana dan kegiatan dalam laboratorium/studio.

b. Fungsi

- Menempatkan bahan dan alat laboratorium komputer
- Menentukan perencanaan pengelolaan laboratorium komputer

BAB III

METODOLOGI PENELITIAN

3.1 Metode Pengumpulan Data

Metode pengumpulan data digunakan penulis untuk mendapatkan data sebagai bahan kajian dalam penulisan tugas akhir dengan tujuan membuat suatu Aplikasi Pendaftaran Mahasiswa Baru Berbasis Android. Dalam hal ini penulis menggunakan metode pengumpulan data berupa sumber data primer (observasi, wawancara) dan sumber data sekunder.

1. Data Primer

Data primer adalah data utama atau data pokok yang digunakan dalam penelitian. Data pokok dapat dideskripsikan sebagai jenis data yang diperoleh langsung dari tangan pertama subjek penelitian atau responden atau informan.

a) Pengamatan (*Observasi*)

Penulis melakukan pengamatan langsung pada AMIK Labuhanbatu Khususnya kepada staff BAAK yang mengurus dan mengatur tentang Akademik AMIK.

b) Wawancara (*Interview*)

Penulis melakukan pendekatan dan mengajukan Tanya jawab langsung kepada Dosen, Staff BAAK dan Mahasiswa-mahasiswi dengan tujuan untuk memperoleh data sesuai dengan tujuan penelitian.

2. Data Sekunder

Data Sekunder adalah data yang diperoleh lewat pihak lain, tidak langsung diperoleh oleh peneliti dari subjek penelitiannya. Data sekunder ini disebut juga dengan Data Tangan Kedua. Data Sekunder biasanya berwujud data dokumentasi atau data laporan yang telah tersedia.

Data diperoleh melalui buku-buku, data-data penelitian, laporan literatur yang berhubungan dengan masalah yang akan diteliti sebagai bahan referensi bagi penulis.

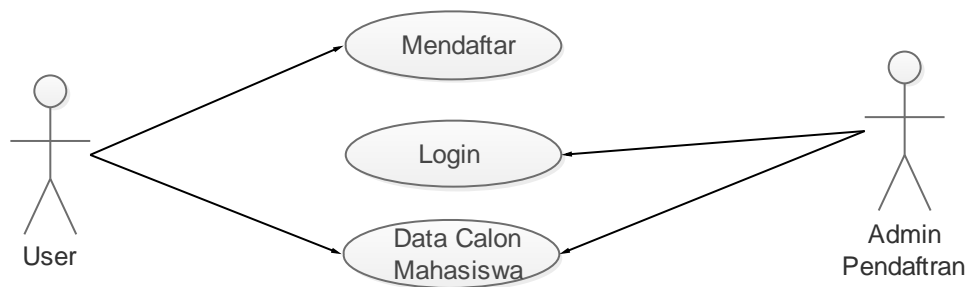
3.2 Metode Perancangan Sistem

Metode perancangan sistem berisi rancangan yang digunakan dalam membangun sistem, diantaranya membangun rancangan *input*, rancangan proses, rancangan *output*, rancangan sistem dan rancangan *interface*.

Rancangan bertujuan untuk memberikan gambaran umum dari sistem yang akan berjalan kepada setiap pengguna. Perancangan adalah sekumpulan aktivitas yang menggambarkan secara rinci bagaimana sistem akan berjalan. Hal itu bertujuan untuk menghasilkan produk perangkat lunak yang sesuai dengan kebutuhan pengguna (*user*). Tahapan perancangan sistem merupakan tahap lanjutan dalam pengembangan sistem, yang dilakukan setelah selesai tahap analisa sistem. Tujuan dari tahapan ini untuk memberikan gambaran kepada *user* tentang bagaimana sistem baru yang diusulkan akan bekerja dan memberikan ilustrasi dan penjelasan yang lengkap kepada *programmer* dalam mengimplementasikan rancangan sistem ke dalam sebuah program aplikasi atau bahasa pemrograman. *UML (Unified Modelling Language)* adalah tahapan-tahapan pekerjaan yang dilakukan oleh analisis sistem dan *programmer* dalam membangun sebuah sistem. Metode-metode *UML* yang digunakan antara lain *use case diagram*, *activity diagram*, *sequence diagram*, dan *component diagram*.

3.2.1 Use Case Diagram

Use Case diagram adalah rangkaian atau uraian sekelompok yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh sebuah *actor*. *Use case diagram* dalam aplikasi pendaftaran mahasiswa baru yaitu :



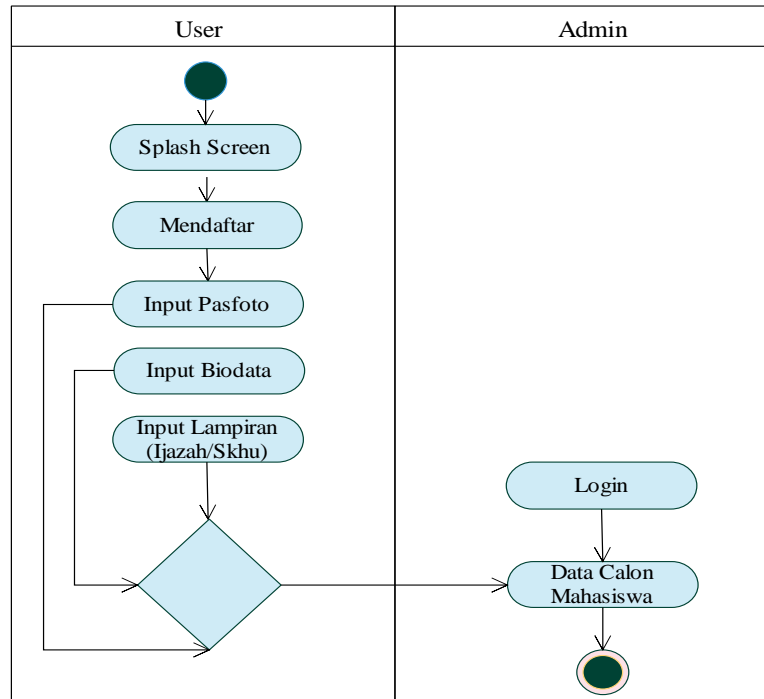
Gambar 3.1 Use Case Diagram

Pada gambar 3.1 *Use Case Diagram* menjelaskan aktivitas yang dapat dilakukan oleh *user* dan *admin* di dalam sistem aplikasi tersebut. *Use case* diatas menjelaskan bahwa *user* dapat Mendaftar sehingga data dapat tersimpan sebagai data calon mahasiswa lalu *admin* dapat melakukan login dan dapat melihat data calon mahasiswa yang sudah mendaftar yang tersimpan didatabase *mysql*.

3.2.2 Activity Diagram

Activity Diagram berfungsi sebagai memodelkan *workflow* proses dan urutan aktivitas dalam suatu proses. Activity diagram dalam aplikasi pendaftaran mahasiswa baru yaitu :

Gambar 3.2 Activity Diagram

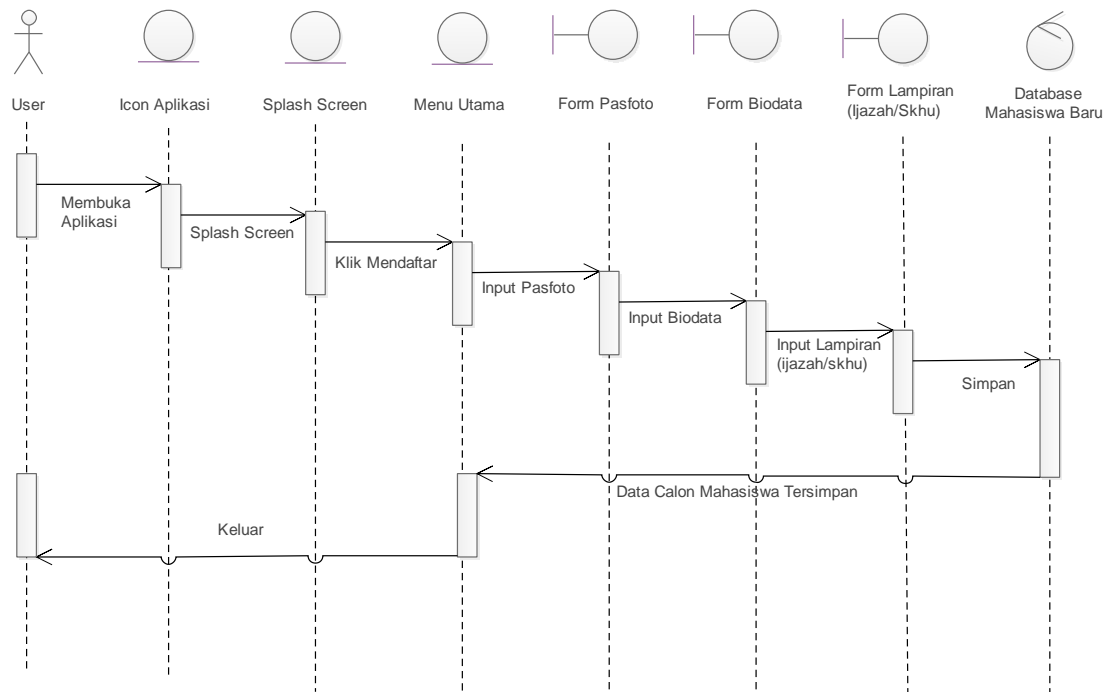


Pada Gambar 3.2 Activity Diagram menjelaskan bahwa :

1. *User* menjalankan Aplikasi dan Aplikasi Menampilkan Splash Screen.
2. *User* Kemudian Melakukan Pendaftaran dengan mengklik Mendaftar.
3. *User* menginputkan Pasfoto, Biodata dan Lampiran (Ijazah/Skhu).
4. Data Yang diinputkan akan tersimpan dan outputnya dapat dilihat di sistem admin.
5. *Admin* melakukan login ke sistem pendaftaran.
6. *Admin* dapat melihat Data calon Mahasiswa yang sudah mendaftar.

3.2.3 Sequence Diagram

Sequence Diagram penggambaran interaksi antar objek didalam dan disekitar sistem kegunaannya untuk menunjukkan rangkaian pesan yang dikirim melalui beberapa objek, juga interaksi antar objek yang terjadi pada titik tertentu dalam eksekusi sistem.



Gambar 3.3 Sequence Diagram

Pada Gambar 3.3 *Sequence Diagram* menjelaskan bahwa :

1. *User* menjalankan Aplikasi dan Aplikasi Menampilkan Splash Screen.
2. *User* Kemudian Melakukan Pendaftaran dengan mengklik Mendaftar.
3. *User* menginputkan Pasfoto, Biodata dan Lampiran (Ijazah/Skhu).
4. Data Yang diinputkan akan tersimpan di database mysql dengan nama mahasiswa baru

3.2.4 Rancangan Masukan (*input*)

Rancangan masukan(*input*) adalah rancangan sebuah form Pasfoto, Biodata dan Lampiran yang dimasukkan ke database *Mysql* kemudian di proses oleh sistem sehingga menghasilkan keluaran(*output*) di dalam *form Admin*. Pada rancangan masukan(*input*) ini terdapat beberapa *form* agar memudahkan *user* dalam menggunakan aplikasi ini. Dengan adanya *form-form* ini maka tampilan dalam sistem Aplikasi ini akan lebih tertata rapi.

1. Form Pasfoto

Form ini merupakan Form yang pertama kali untuk memulai proses Pendaftaran di dalam Aplikasi Pendaftaran Mahasiswa Baru.



Gambar 3.4 Rancangan *Form Pasfoto*

Nama Masukan	: Form Pasfoto
Fungsi	: Untuk menginput Pasfoto ke database
mysql	
Media	: <i>Smartphone Android</i>
Distribusi	: <i>User</i>
Keterangan	: Pasfoto yang ditambahkan harus
format.jpg,	
	.jpeg, .png

2. Form Biodata

Form ini merupakan Form selanjutnya dari form pasfoto di dalam Aplikasi Pendaftaran Mahasiswa Baru.

Biodata Diri

ID PENDAFTARAN

NAMA LENGKAP

JENIS KELAMIN

ALAMAT

AGAMA

NO.HANDPHONE

ASAL SEKOLAH

TAHUN LULUS

SIMPAN BIODATA

Gambar 3.5 Rancangan Form Biodata

Nama Masukan : Form Biodata
Fungsi : Untuk menginput Biodata diri calon mahasiswa ke database mysql
Media : *Smartphone Android*
Distribusi : *User*

3. Form Lampiran(Ijazah/Skhu)

Form ini merupakan Form selanjutnya dari form biodata dan

Lampiran: Ijazah/Skhu

ID PENDAFTARAN

Pilih Foto

Preview Foto

li

UPLOAD

merupakan form terakhir di dalam Aplikasi Pendaftaran Mahasiswa Baru.

Gambar 3.6 Rancangan *Form Lampiran*

Nama Masukan : Form Lampiran (ijazah/skhu)
 Fungsi : Untuk menginput ijazah/skhu ke database
 mysql
 Media : *Smartphone Android*
 Distribusi : *User*
 Keterangan : Ijazah/Skhu yang ditambahkan harus di
 scan
 terlebih dahulu dan dalam format.jpg,
 .jpeg, .png

3.2.5 Rancangan Proses (*Process*)

Dalam pembuatan sistem alat yang digunakan dalam membuat perancangan dan desain aplikasi pendaftaran mahasiswa baru yaitu dengan menggunakan *Unified Modeling Language (UML)*. *Unified Modelling Language* adalah sebuah bahasa yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak.

3.2.6 Rancangan Keluaran (*Output*)

Rancangan keluaran(*output*) adalah rancangan sebuah form tabel data calon mahasiswa baru yang merupakan output dari data-data yang telah diinputkan *user* pada aplikasi pendaftaran mahasiswa baru.

Tabel 3.1 Rancangan Output Form Tabel Data Calon Mahasiswa Baru

No	Id Pendaftaran	Pasfoto	Nama Lengkap	Jenis Kelamin	Tempat/Tanggal Lahir	alamat	agama	No.Hp	Asal Sekolah	Tahun Lulus	Lampiran (Ijazah/Skhu)
9[11]	X[24]	X[560]	X[560]	X[24]	X[50]	X[225]	X[23]	X[14]	X[150]	X[10]	X[560]

Nama Keluaran : Form Tabel Data Calon Mahasiswa Baru

Fungsi sudah baru : Untuk menampilkan *output* dari data-data yang diinputkan pada aplikasi pendaftaran mahasiswa baru

Media : *Web*

Distribusi : *Admin*

Keterangan mahasiswa : Output Keseluruhan dari data para calon mahasiswa

3.2.7 Rancangan Basis Data

Rancangan basis data(database) berisikan proses menciptakan perancangan suatu basis data (database) yang akan mendukung operasi tujuan dalam merancang suatu basis data yang digunakan dalam perancangan sistem aplikasi pendaftaran mahasiswa baru. Berikut adalah rancangan basis data yang digunakan :

Nama File : database mahasiswabaruu

Media : *MySQL*

Tabel 3.2 Tabel DataBase Mahasiswa Baru

No	Nama Tabel	Keterangan
1	admin	Berisi tentang Data Admin yang bisa login ke Sistem
2	biodata	Data Biodata Diri Calon Mahasiswa
3	ijazah	Data Lampiran (Ijazah/skhu)
4	pasfoto	Data Pas Foto

3.2.7.1 Normalisasi

Normalisasi merupakan proses pengelompokan elemen data menjadi tabel-tabel yang menunjukkan entity dan relasinya. Pada dasarnya normalisasi adalah suatu teknik menstruktur data dalam cara-cara tertentu untuk membantu

mengurangi atau mencegah timbulnya masalah yang berhubungan dengan pengolahan data dalam database.

1. Bentuk tidak normal

Bentuk ini merupakan kumpulan data yang akan direkam, tidak ada keharusan mengikukti format tertentu, dapat saja data tidak lengkap atau terduplikasi. Data dikumpulkan apa adanya sesuai dengan saat menginput.

Tabel 3.3 Tabel Bentuk Tidak Normal

No	Id Pendaftaran	Pasfoto	Nama Lengkap	Jenis Kelamin	Tempat/Tanggal Lahir	alamat	agama	No.Hp	Asal Sekolah	Tahun Lulus	Lampiran (Ijazah/Skhu)
1	AP190598	X[560]	Arie Pramana	Laki-laki	Sisumut, 19 Mei 1998	Sisumut	Islam	0852 7531 4543	Sman 1 Kotapinang	2016	X[560]
2	IFL061098	X[560]	Indah Frisky Ladevi	Perempuan	Air Serdang, 06 Oktober 1998		Islam	0822 7808 2019		2016	X[560]
3	AY280198	X[560]	Alfitra Yanuar	Laki-laki		Cikampak		0813 7721 1075		2016	X[560]

2. Bentuk Normal Pertama (1NF)

Bentuk normal pertama adalah terpenuhinya suatu table yang tidak memiliki atribut bernilai banyak atau lebih dari satu atribut dengan domain nilai yang sama, dalam bentuk normal pertama data tidak mengandung atribut yang bernilai ganda (hanya mengandung 1 nilai) dan harus diisi datanya (tidak boleh kosong).

Tabel 3.4 Tabel Bentuk Normal Pertama

No	Id Pendaftaran	Pasfoto	Nama Lengkap	Jenis Kelamin	Tempat/Tanggal Lahir	alamat	agama	No.Hp	Asal Sekolah	Tahun Lulus	Lampiran (Ijazah/Skhu)
1	AP190598	X[560]	Arie Pramana	Laki-laki	Sisumut, 19 Mei 1998	Sisumut	Islam	0852 7531 4543	Sman 1 Kotapinang	2016	X[560]
2	IFL061098	X[560]	Indah Frisky Ladevi	Perempuan	Air Serdang, 06 Oktober 1998	Air Serdang	Islam	0822 7808 2019	Smk Swasta Sitibanun Sigambal	2016	X[560]
3	AY280198	X[560]	Alfitra Yanuar	Laki-laki	Cikampak, 28 Januari 1998	Cikampak	Islam	0813 7721 1075	Smk Swasta Indonesia Membangun 2	2016	X[560]

3. Bentuk Normal Kedua (2NF)

Normalisasi kedua (2NF), suatu relasi memenuhi relasi kedua jika dan hanya jika relasi tersebut memenuhi normal pertama dan setiap atribut yang bukan kunci (non key) bergantung secara fungsional terhadap kunci utama (Primary key).

a) Tabel Biodata

Tabel 3.5 Tabel Biodata

Kd_mhs	Id Pendaftaran	Nama Lengkap	Jenis Kelamin	Tempat/Tanggal Lahir	alamat	agama	No.Hp	Asal Sekolah	Tahun Lulus
1	AP190598	Arie Pramana	Laki-laki	Sisumut, 19 Mei 1998	Sisumut	Islam	0852 7531 4543	Sman 1 Kotapinang	2016
2	IFL061098	Indah Frisky Ladevi	Perempuan	Air Serdang, 06 Oktober 1998	Air Serdang	Islam	0822 7808 2019	Smk Swasta Sitibanun Sigambal	2016
3	AY280198	Alfitra Yanuar	Laki-laki	Cikampak, 28 Januari 1998	Cikampak	Islam	0813 7721 1075	Smk Swasta Indonesia Membangun 2	2016

b) Tabel Pasfoto

Tabel 3.6 Tabel pasfoto

id_foto	Id Pendaftaran	Pasfoto
1	AP190598	X[560]
2	IFL061098	X[560]
3	AY280198	X[560]

c) Tabel Ijazah

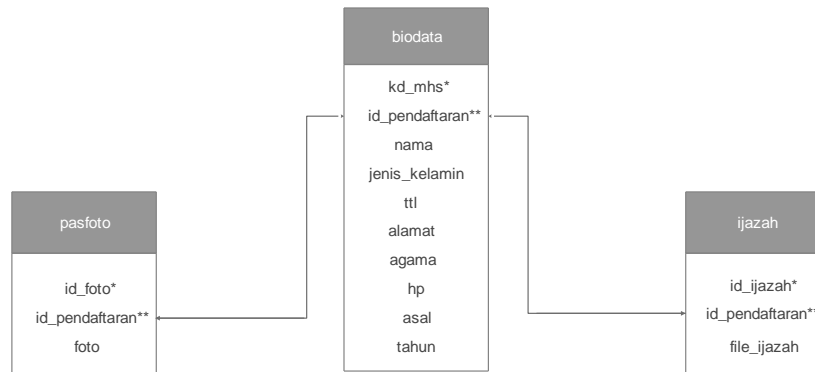
Tabel 3.7 Tabel ijazah

id_ijazah	Id Pendaftaran	file_ijazah
1	AP190598	X[560]
2	IFL061098	X[560]
3	AY280198	X[560]

3.2.7.2 Relasi Tabel

Relasi Tabel secara sederhana dapat dikatakan sebagai suatu database yang didalamnya terdapat tabel-tabel yang saling berelasi satu sama lain.

Gambar 3.7 Relasi Tabel



Keterangan tanda pada Gambar 3.7 Relasi Tabel

* : Primary key

** : Foreign key

3.2.7.3 Desain File/Tabel

Desain tabel berisi *kumpulan* dari *field-field* yang terdapat pada table didalam *database*, antara lain nama *file*, *primarykey*, yang terdapat pada *file* tersebut, dan struktur data-data yang terdapat pada *file*. Berikut ini *file-file*/Tabel yang digunakan dalam aplikasi pendaftaran mahasiswa baru.

1. Tabel admin.

Nama File : admin

Media : *MySQL*

Primary key : *id*

Tabel 3.8 Struktur Tabel Admin

No	Field	Type	Size	Keterangan
1	id	Integer	5	Primary Key
2	nama	Varchar	200	Nama Lengkap Admin
3	username	Varchar	150	username yang admin gunakan
4	password	Varchar	220	password yang admin gunakan

2. Tabel biodata.

Nama File : biodata

Media : *MySQL*

Primary key : *kd_mhs*

Tabel 3.9 Struktur Tabel Biodata

No	Field	Type	Size	Keterangan
1	kd_mhs	Integer	11	Primary Key
2	id_pendafran	Varchar	24	Id pendaftaran Calon Mahasiswa
3	nama	Varchar	560	Nama Lengkap Calon Mahasiswa
4	jenis_kelamin	Varchar	24	Jenis Kelamin dari Calon Mahasiswa
5	ttl	Varchar	50	Tempat/Tanggal lahir Calon Mahasiswa
6	alamat	Varchar	225	Alamat Calon Mahasiswa
7	agama	Varchar	23	Agama Calon Mahasiswa
8	hp	Varchar	14	Nomer Handphone Calon Mahasiswa
9	asal	Varchar	150	Asal Sekolah Calon Mahasiswa
10	tahun	Varchar	10	Tahun Lulus Calon Mahasiswa

3. Tabel lampiran(Ijazah/Skhu).

Nama File : ijazah

Media : *MySQL*

Primary key : *id_ijazah*

Tabel 3.10 Struktur Tabel Ijazah

No	Field	Type	Size	Keterangan
1	id_ijazah	Integer	11	Primary Key
2	id_pendafran	Varchar	24	Id Pendaftarann Calon Mahasiswa
3	file_ijazah	Varchar	560	File atau foto Ijazah/Skhu Calon Mahasiswa

4. Tabel Pasfoto.

Nama File : pasfoto

Media : *MySQL*

Primary key : *id_ijazah*

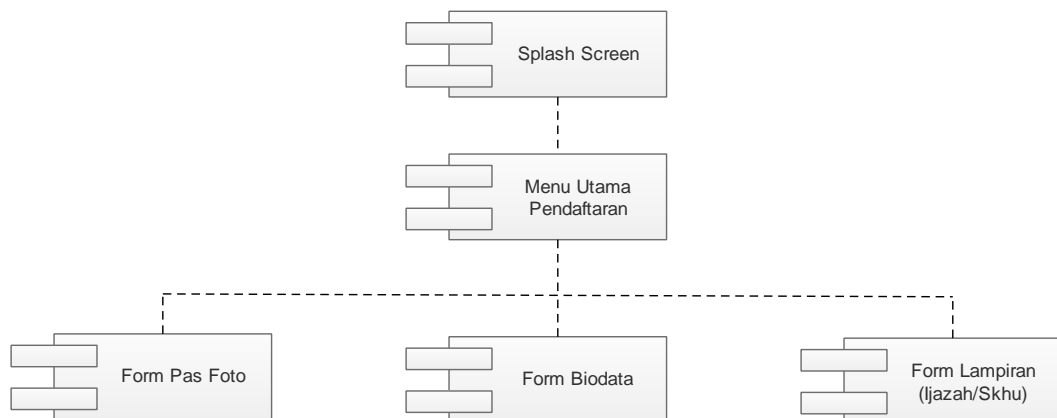
Tabel 3.11 Struktur Tabel Pasfoto

No	Field	Type	Size	Keterangan
1	id_foto	Integer	11	Primary Key
2	id_pendafran	Varchar	24	Id Pendaftarann Calon Mahasiswa
3	foto	Varchar	560	Pasfoto Calon Mahasiswa

3.2.8 Rancangan Interface

Gambar rancangan *interface* Aplikasi Pendaftaran Mahasiswa Baru (Studi Kasus: AMIK Labuhanbatu) memiliki 2 interface yaitu :

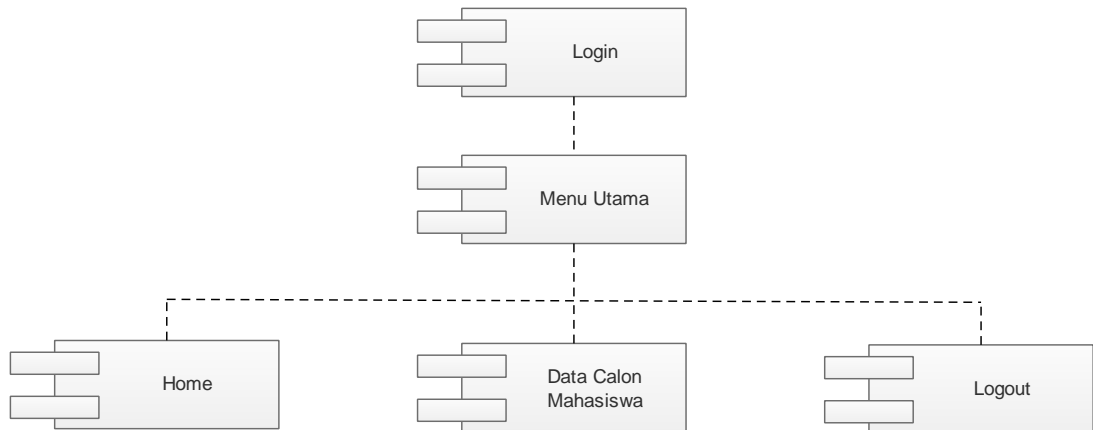
1) *Interface User*



Gambar 3.8 Component Diagram User

Pada gambar 3.8. *Component Diagram* user dari aplikasi pendaftaran mahasiswa baru menjelaskan rancangan pada sistem terdapat *Splash Screen* Untuk memulai aplikasi, lalu masuk ke menu utama pendafrtran, didalamnya terdapat *form pasfoto*, *form biodata* dan *form lampiran(ijazah/skhu)* yang bertujuan menginputkan data pendaftar ke database *mysql*.

2) *Interface Admin*



Gambar 3.9 *Component Diagram Admin*

Pada gambar 3.9 *Component Diagram Admin* aplikasi pendaftaran mahasiswa baru menjelaskan rancangan pada sistem terdapat *Login* Untuk memulai sistem agar rahasia data terjaga, lalu masuk ke menu utama home, lalu terdapat menu data calon mahasiswa, dan juga *logout*.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Implementasi

Implementasi adalah penerapan cara kerja sistem berdasarkan hasil analisa dan juga perancangan yang telah dibuat sebelumnya ke dalam suatu bahasa pemrograman tertentu.

Tahap implementasi merupakan tahap penciptaan perangkat lunak, tahap kelanjutan dari kegiatan perancangan sistem. Tahap ini merupakan tahap dimana sistem siap untuk dioperasikan, yang terdiri dari penjelasan mengenai lingkungan implementasi, implementasi *database* dan implementasi program.

Adapun pembahasan implementasi terdiri dari perangkat keras pembangun, perangkat lunak pembangun, dan implementasi tampilan/antarmuka (*interface*).

4.1.1 Perangkat Keras

Berikut adalah spesifikasi perangkat yang digunakan saat merancang dan membangun Aplikasi Pendaftaran Mahasiswa Baru (Studi Kasus : AMIK Labuhanbatu), yaitu :

1. Processor Intel Pentium 987, 1.5GHz 2MB
2. Chipset Intel HM70
3. Memory 2048MB RAM
4. Hardisk 500GB
5. Monitor 14"
6. Keyboard & Mouse

4.1.2 Perangkat Lunak

Perangkat lunak yang digunakan untuk mengimplementasikan Aplikasi Pendaftaran Mahasiswa Baru (Studi Kasus : AMIK Labuhanbatu) ini adalah sebagai berikut :

1. *Personal Computer*

Perangkat lunak yang dibutuhkan pada *personal computer* yaitu :

- a. Sistem Operasi *Windows 7*

- b. *Eclipse*
- c. *ADT v22.0.5-757759*
- d. *Android SDK*
- e. *Java development kit*
- f. Web Browser (Mozilla Firefox/Google Chrome)
- g. XAMPP
- h. Sublime Text3

2. *Smartphone*

Perangkat lunak yang dibutuhkan pada *smartphone Samsung Galaxy J7 Prime* untuk mengimplementasikan Aplikasi Pendaftaran Mahasiswa Baru (Studi Kasus : AMIK Labuhanbatu) ini yaitu :

- a. Nomer model SM-G610F/DS
- b. Versi *android* 8.1.0
- c. CPU *Octa-core* 1.6 GHz Cortex-A53
- d. RAM 3 GB
- e. Memori Internal 32 GB

4.1.3 Implementasi *Database*

Pembuatan *database* dilakukan dengan menggunakan aplikasi *XAMPP* yang merupakan *DBMS Mysql*. Implementasi *database* adalah sebagai berikut :

1. Tabel Admin

Implementasi dari rancangan basis data tabel admin dapat dilihat pada gambar 4.1.

Gambar 4.1 Tabel Admin

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	id	int(5)			No	None	AUTO_INCREMENT
2	nama	varchar(200)	latin1_swedish_ci		No	None	
3	username	varchar(150)	latin1_swedish_ci		No	None	
4	password	varchar(220)	latin1_swedish_ci		No	None	

2. Tabel Biodata

Implementasi dari rancangan basis data tabel biodata dapat dilihat pada gambar 4.2.

Gambar 4.2 Tabel Biodata

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	kd_mhs	int(11)			No	None	AUTO_INCREMENT
2	id_pendaftaran	varchar(24)	latin1_swedish_ci		No	None	
3	nama	varchar(560)	latin1_swedish_ci		No	None	
4	jenis_kelamin	varchar(24)	latin1_swedish_ci		No	None	
5	tgl	varchar(50)	latin1_swedish_ci		No	None	
6	alamat	varchar(225)	latin1_swedish_ci		No	None	
7	agama	varchar(23)	latin1_swedish_ci		No	None	
8	hp	varchar(14)	latin1_swedish_ci		No	None	
9	asal	varchar(150)	latin1_swedish_ci		No	None	
10	tahun	varchar(10)	latin1_swedish_ci		No	None	

3. Tabel Ijazah

Implementasi dari rancangan basis data tabel ijazah dapat dilihat pada gambar 4.3.

Gambar 4.3 Tabel Ijazah

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	id_ijazah	int(11)			No	None	AUTO_INCREMENT
2	id_pendaftaran	varchar(24)	latin1_swedish_ci		No	None	
3	file_ijazah	varchar(560)	latin1_swedish_ci		No	None	

4. Tabel Pasfoto

Implementasi dari rancangan basis data tabel pasfoto dapat dilihat pada gambar 4.4.

Gambar 4.4 Tabel Pasfoto

#	Name	Type	Collation	Attributes	Null	Default	Extra
1	<u>id_foto</u>	int(11)			No	None	AUTO_INCREMENT
2	<u>id_pendaftaran</u>	varchar(24)	latin1_swedish_ci		No	None	
3	foto	varchar(560)	latin1_swedish_ci		No	None	

4.1.4 Implementasi Tampilan

Implementasi tampilan/antarmuka dilakukan dengan setiap halaman aplikasi yang dibuat dan pengkodeannya dalam bentuk file program. Implementasi tampilan Aplikasi Pendaftaran Mahasiswa Baru (Studi Kasus : AMIK Labuhanbatu) ini Terdiri dari 2 Tampilan/Interface yaitu tampilan *user* dan tampilan admin. Berikut ini adalah implementasi antarmuka yang dibuat.

4.1.4.1 Tampilan *User*

Tampilan *user* merupakan tampilan aplikasi yang digunakan *user* di dalam *Smartphone* Berbasis Android. Tampilan *user* memiliki beberapa Tampilan yang terdapat di dalam Aplikasi Pendaftaran Mahasiswa Baru (Studi Kasus : AMIK Labuhanbatu). Tampilan *user* adalah sebagai berikut :

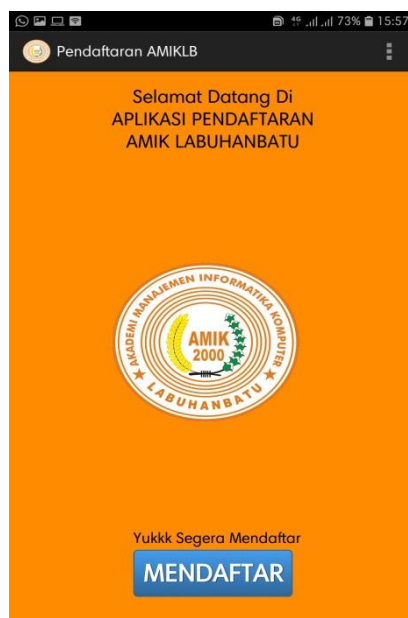
1. Tampilan *Splashscreen*



Gambar 4.5 *Splashscreen*

Pada gambar 4.5 merupakan tampilan *splashscreen* yang ada di Aplikasi Pendaftaran Mahasiswa Baru Berbasis Android (Studi Kasus : AMIK Labuhanbatu) sebelum masuk ke menu utama. Di dalam *splashscreen* ini terdapat Gambar Icon dari AMIK Labuhanbatu, bacaan aplikasi pendaftaran mahasiswa baru AMIK Labuhanbatu, icon *loading* serta tulisan tunggu sebentar.

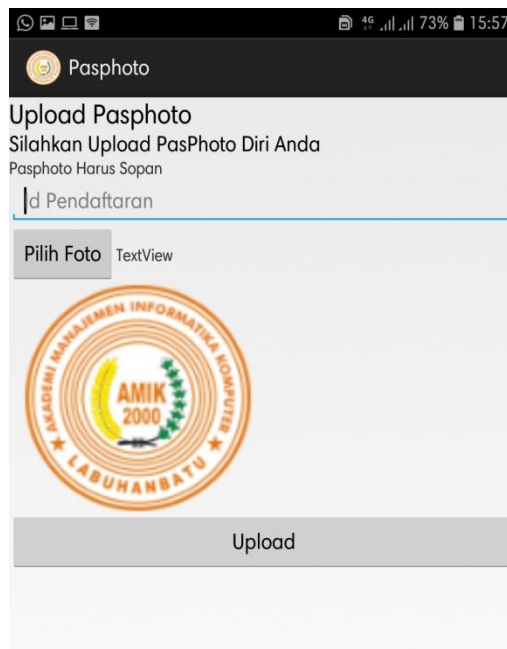
2. Tampilan Menu Utama



Gambar 4.6 Menu Utama

Pada gambar 4.6 merupakan tampilan menu utama yang ada di Aplikasi Pendaftaran Mahasiswa Baru Berbasis Android (Studi Kasus : AMIK Labuhanbatu) Di dalam menu utama ini terdapat Gambar Icon dari AMIK Labuhanbatu, bacaan selamat datang di aplikasi pendaftaran AMIK Labuhanbatu, tulisan yukkk segera mendaftar dan tombol “MENDAFTAR”.

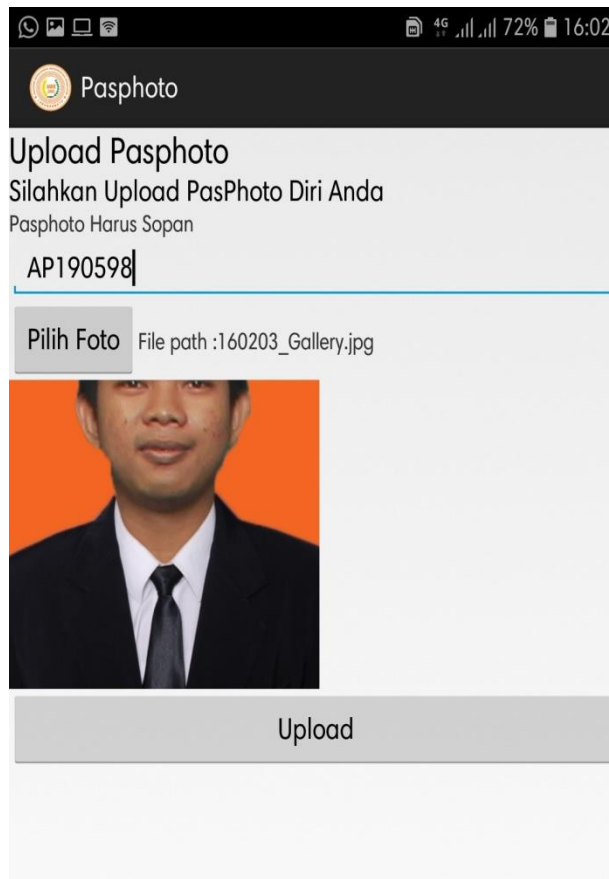
3. Tampilan Input Pasfoto



Gambar 4.7 Form Pasfoto

Pada gambar 4.7 merupakan tampilan form menginput pasfoto. Di dalam form ini terdapat *Textfield* yang berfungsi sebagai penginputan “Id Pendaftaran” Tombol “Pilih Foto” tombol yang membawa ke galeri untuk memilih foto yang akan di upload, Tulisan “*TextView*” nanti akan berganti

dengan otomatis setelah Foto telah terpilih menjadi nama file dari foto yang dipilih tadi, Gambar Icon dari AMIK Labuhanbatu yang merupakan preview foto yang akan diupload dan icon tersebut akan berganti otomatis menjadi foto yang akan diupload, dan Tombol “Upload” untuk menyimpan



penginputan pasfoto.

Gambar 4.8 Contoh Input Pasfoto

Pada gambar 4.8 merupakan contoh penginputan pasfoto. Penginputan dimulai dengan mengetik Id Pendaftaran “AP190598” di *textfield*. Dengan Nama File Foto yang diupload “160203_Gallery.jpg”, Dan ada Preview Foto dari pasfoto yang akan diupload. Selanjutnya *user* mengklik tombol “Upload” dan penginputan pasfoto akan diproses dan tersimpan.

4. Tampilan Input Biodata



The image shows a mobile application interface for entering personal data. At the top, there is a status bar with icons for signal, Wi-Fi, and battery (72%), and the time 16:03. Below the status bar is a header with a circular icon and the text 'Biodata Diri'. The main title of the form is 'Masukkan Biodata Diri Anda'. The form contains several text input fields: 'ID Pendaftaran', 'Nama Lengkap', 'Jenis Kelamin (Laki-laki/Perempuan)', 'Tempat/Tanggal Lahir', 'Agama', 'Alamat', 'No Handphone', 'Asal Sekolah', and 'Tahun Tamat'. At the bottom of the form is a grey button labeled 'Simpan Biodata Diri'.

Gambar 4.9 Form Biodata

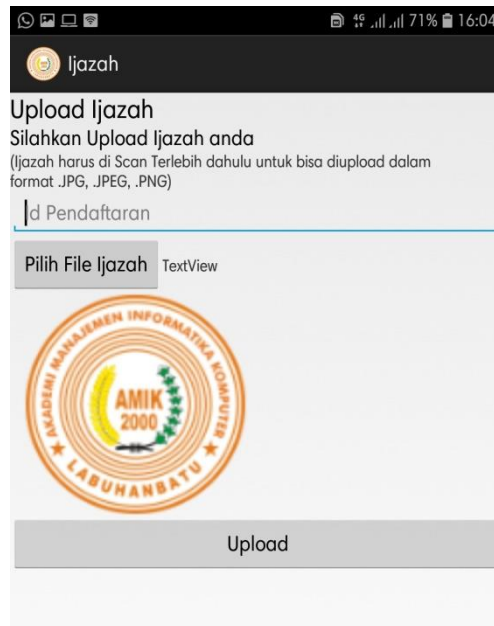
Pada gambar 4.9 merupakan tampilan form menginput biodata. Di dalam form ini terdapat hampir semuanya *Textfield* sebagai penginputan biodata. Biodata yang akan diinput yaitu “Id Pendaftaran, Nama Lengkap, Jenis Kelamin, Tempat/Tanggal Lahir, Agama, Alamat, No. Handphone, Asal Sekolah, Tahun Tamat ” terdapat juga Tombol “Simpan Biodata Diri” untuk menyimpan penginputan biodata.

The image shows a mobile application interface for entering personal data. The title bar at the top reads 'Biodata Diri'. Below it, the instruction 'Masukkan Biodata Diri Anda' is displayed. The form consists of ten text input fields, each containing a specific piece of information: 'AP190598', 'Arie Pramana', 'Laki-laki', 'Sisumut, 19 Mei 1998', 'Islam', 'Sisumut', '0852 7531 4543', 'Sman 1 Kotapinang', and '2016'. At the bottom of the form is a grey button labeled 'Simpan Biodata Diri'. The status bar at the top of the phone shows the time as 16:04, 71% battery, and 4G connectivity.

Gambar 4.10 Contoh Input Biodata

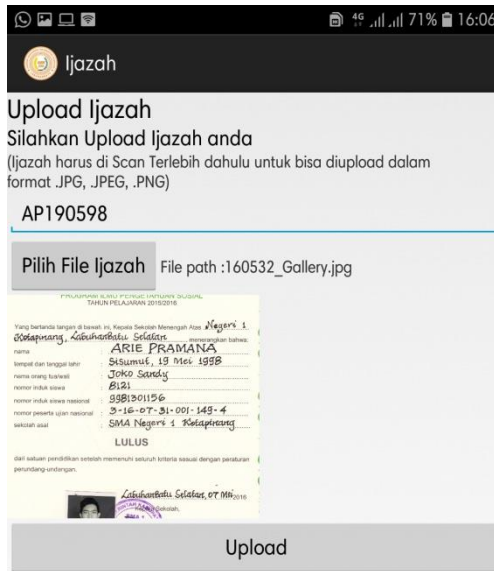
Pada gambar 4.10 merupakan contoh penginputan biodata. Penginputan keseluruhan menetik di *textfield*. Pada Gambar 4.10 di inputkan Id pendaftaran “AP190598”, nama lengkap “Arie Pramana”, jenis kelamin “Laki-laki”, tempat/tanggal lahir “Sisumut, 19 Mei 1998”, agama “Islam”, alamat “Sisumut” no. handphone “0852 7531 4543” asal sekolah ”Sman 1 Kotapinang” dan tahun tamat “2016”. Kemudian *User* mengklik tombol “Simpan Biodata Diri” untuk menyimpan penginputan biodata.

5. Tampilan Input Ijazah



Gambar 4.11 Form Lampiran

Pada gambar 4.11 merupakan tampilan form menginput lampiran berupa Ijazah/Skhu. Di dalam form ini terdapat *Textfield* yang berfungsi sebagai penginputan “Id Pendaftaran” Tombol “Pilih File Ijazah” tombol yang membawa ke galeri untuk memilih foto ijazah yang akan di upload, Tulisan “*TextView*” nanti akan berganti dengan otomatis setelah Foto ijazah telah terpilih menjadi nama file dari foto ijazah yang dipilih tadi, Gambar Icon dari AMIK Labuhanbatu yang merupakan preview foto ijazah yang akan diupload dan icon tersebut akan berganti otomatis menjadi foto ijazah yang akan diupload, dan Tombol “Upload” untuk menyimpan penginputan lampiran(ijazah/skhu).



Gambar 4.12 Contoh Input Lampiran

Pada gambar 4.12 merupakan contoh penginputan lampiran Ijazah. Penginputan dimulai dengan mengetik Id Pendaftaran “AP190598” di *textfield*. Dengan Nama File ijazah yang diupload “160532_Gallery.jpg”, Dan ada Preview Foto ijazah dari ijazah yang akan diupload. Selanjutnya *user* mengklik tombol “Upload” dan penginputan lampiran(ijazah/skhu) akan diproses dan tersimpan.

4.1.4.2 Tampilan Admin

Tampilan *Admin* merupakan tampilan aplikasi yang digunakan *admin* di *Web* untuk *admin* dapat melihat data dari *user* yang sudah melakukan pendaftaran di Aplikasi Pendaftaran Mahasiswa Baru (Studi Kasus : AMIK Labuhanbatu). Tampilan *admin* adalah sebagai berikut :

1. Tampilan Login



ADMIN AMIK LABUHANBATU LOGIN
UNTUK MENGAkses DATA
MASUKKAN USERNAME DAN PASSWORD

User name	Password
-----------	----------

LOGIN

Gambar 4.13 Tampilan Login Admin

Pada gambar 4.13 merupakan tampilan login admin. Admin harus menginputkan *Username* dan *password* dengan benar jika tidak maka admin tidak bisa masuk ke sistem aplikasi.



ADMIN AMIK LABUHANBATU LOGIN
UNTUK MENGAkses DATA
MASUKKAN USERNAME DAN PASSWORD

AMIK LABUHANBATU	●●●●●●●●
------------------	----------

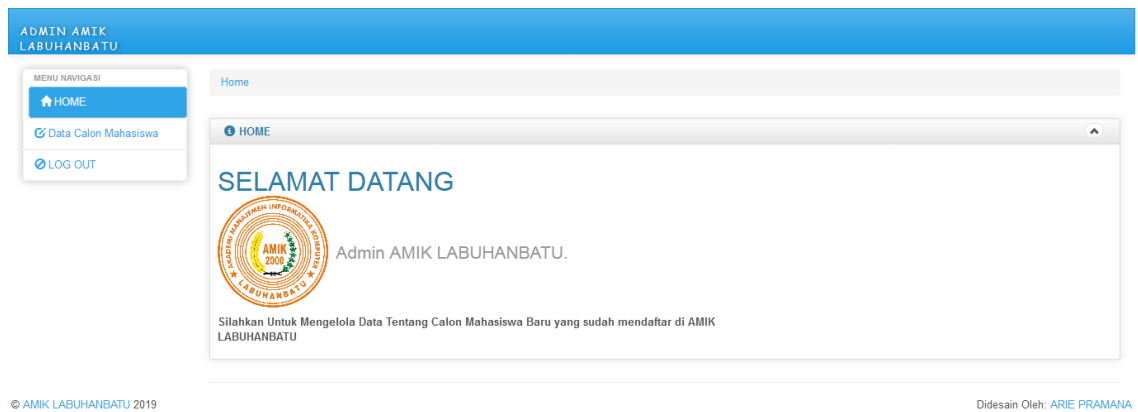
LOGIN

Gambar 4.14 Input Login Admin

Pada gambar 4.14 Admin menginputkan username “AMIK LABUHANBATU” dan password ”semakinmaju” akan tetapi password tidak

terlihat karena demi keamanan data admin. Admin Kemudian Mengklik tombol Login dan admin berhasil masuk ke sistem.

2. Tampilan Home



Gambar 4.15 Tampilan Home

Pada gambar 4.15 adalah Tampilan *Home*. *Home* adalah halaman awal yang akan tampil setelah admin berhasil login di sistem admin aplikasi pendaftaran mahasiswa baru. Di *home* akan menampilkan sedikit tentang ucapan selamat datang kepada admin dan menampilkan logo icon dari AMIK Labuhanbatu. Di sistem admin terdapat 3 navigasi menu yaitu *home*, data calon mahasiswa, dan *logout*.

3. Tampilan Data Calon Mahasiswa

ADMIN AMIK LABUHANBATU

MENU NAVIGASI

- HOME
- Data Calon Mahasiswa
- LOG OUT

Home / Daftar Calon Mahasiswa

Data Calon Mahasiswa

No	ID Pendaftaran	Pasfoto	Nama Lengkap	Jenis Kelamin	Tempat/Tanggal Lahir	Alamat	Agama	No.Handphone	Asal Sekolah	Tahun Tamat	Lampiran(Ijazah)
1	AP190598		Arie Pramana	Laki-laki	Sisumut, 19 Mei 1998	Sisumut	Islam	0852 7531 4543	Sman 1 Kotapinang	2016	 Perbesar Ijazah
2	IFL061088		Indah Frisky Ladevi	Perempuan	Air Serdang, 06 Oktober 1998	Air Serdang	Islam	0822 7808 2019	Smk Swasta Sibunan Sigambal	2016	 Perbesar Ijazah
3	AY280198		Alfira Yanuar	Laki-laki	Cikampak, 28 Januari 1998	Cikampak	Islam	0813 7721 1075	Smk Swasta Indonesia Membangun 2	2016	 Perbesar Ijazah

© AMIK LABUHANBATU 2018

Didesain Oleh: ARIE PRAMANA

Gambar 4.16 Tampilan Data Calon Mahasiswa

Pada gambar 4.16 adalah Tampilan Data Calon Mahasiswa. Data calon mahasiswa adalah halaman untuk melihat data keseluruhan calon mahasiswa yang mendaftar di Aplikasi Pendaftaran Mahasiswa Baru Berbasis Android (Studi Kasus : AMIK Labuhanbatu). Di halaman ini admin dapat melihat keseluruhan data dari pasfoto, biodata dan juga lampiran (ijazah/skhu) calon mahasiswa. Sebagai contoh data yaitu dengan no “1” , id pendaftaran “AP190598” pasfoto terlihat pada gambar 4.16 nama lengkap “Arie Pramana”, jenis kelamin “Laki-laki”, tempat/tanggal lahir “Sisumut, 19 Mei 1998”, alamat “Sisumut” , agama “Islam” no. handphone “0852 7531 4543” asal sekolah ”Sman 1 Kotapinang”, tahun tamat “2016” dan lampiran(ijazah/skhu) terlihat pada gambar 4.16 juga terdapat tombol dengan bertuliskan”Perbesar Ijazah” di kolom

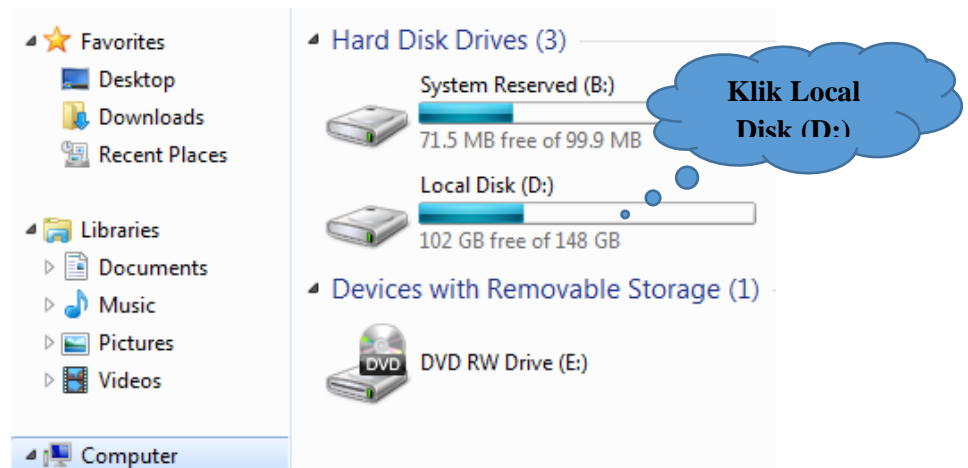
lampiran(ijazah/skhu) untuk admin dapat memperbesar file foto ijazah dari calon mahasiswa tersebut.

4.2 Pengujian

Pengujian dilakukan bertujuan untuk menemukan kesalahan-kesalahan atau kekurangan-kekurangan pada sistem aplikasi yang diuji. Pengujian bermaksud untuk mengetahui sistem aplikasi yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan sistem aplikasi tersebut. Pengujian dilakukan pada *smartphone Samsung Galaxy J7 Prime*, langkah – langkahnya sebagai berikut :

1. File Aplikasi pendaftaran AMIK Labuhanbatu

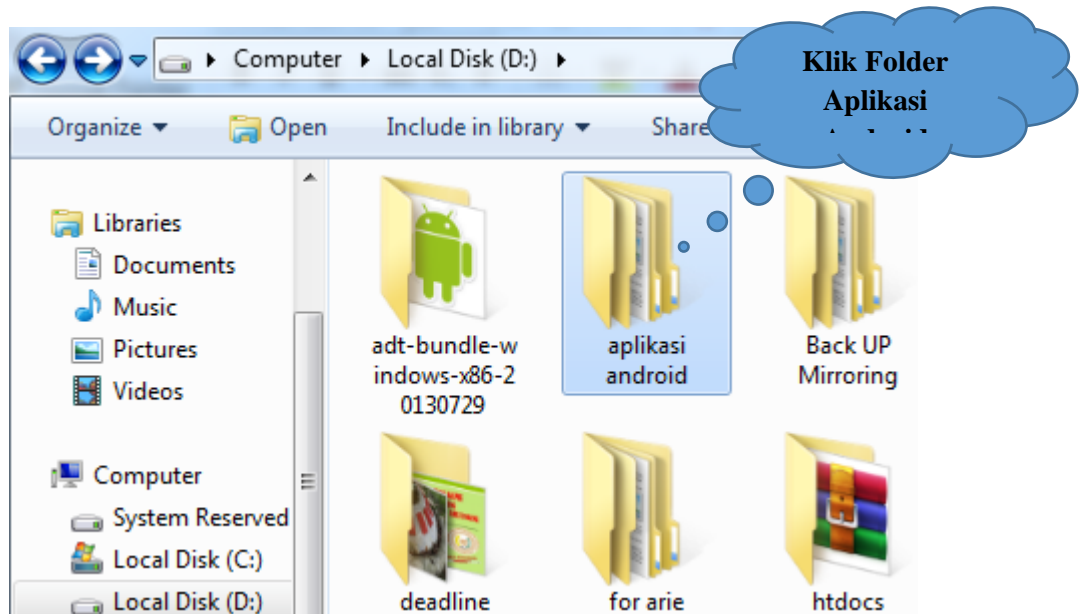
a. Local Disk D



Gambar 4.17 Local Disk D

Pada gambar 4.17 merupakan tampilan *Local Disk D*, disini *user* dapat mengklik folder *Local Disk D* untuk memulai proses pengcopian *file* APK aplikasi pendaftaran mahasiswa baru berbasis *android* (studi kasus : AMIK Labuhanbatu).

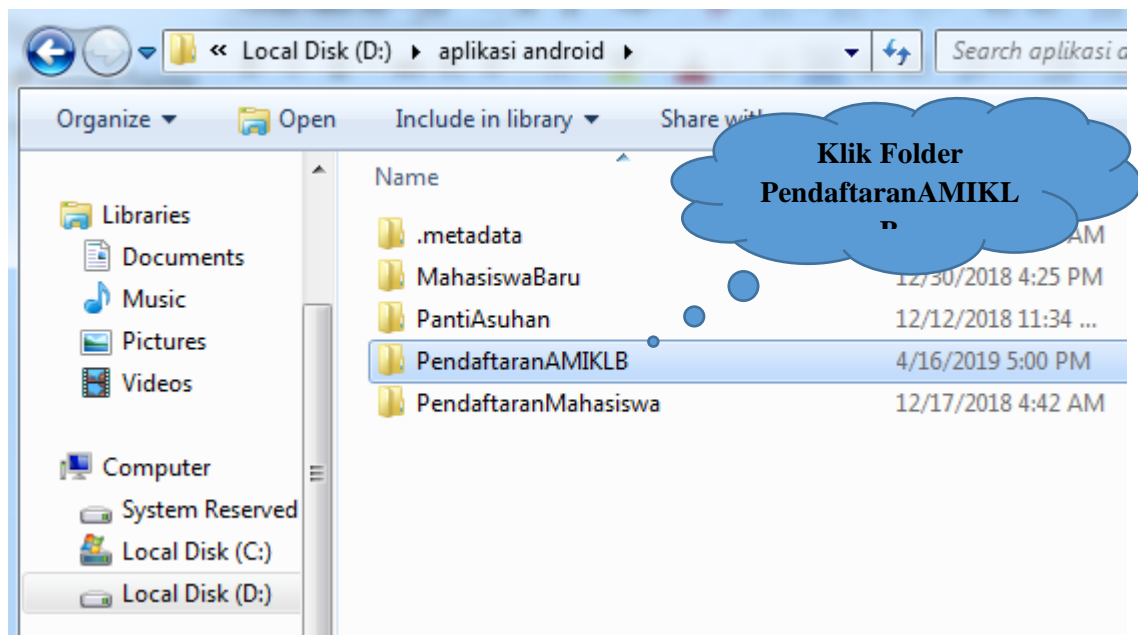
b. Folder Aplikasi Android



Gambar 4.18 Folder Aplikasi *Android*

Pada gambar 4.18 merupakan tampilan folder aplikasi *android*, disini merupakan folder penyimpanan project program-program *android* dan *user* dapat mengklik folder aplikasi *android* untuk melakukan pengcopian *file* APK.

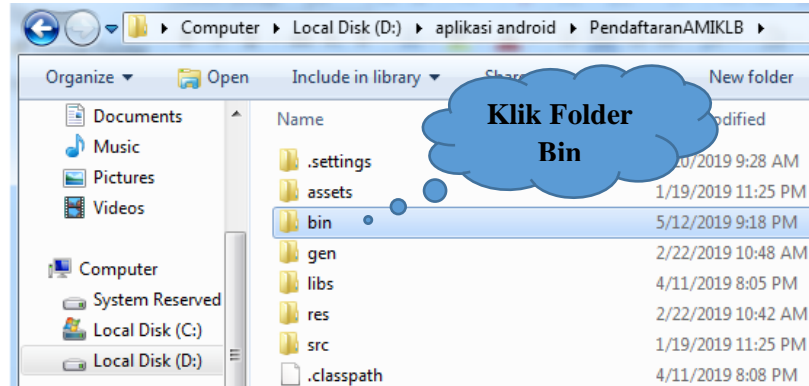
c. Folder PendaftaranAMIKLB



Gambar 4.19 Folder Pendaftaran AMIKLB

Pada gambar 4.19 merupakan folder Pendaftaran AMIK Labuhanbatu disini *user* dapat mengklik folder PendaftaranAMIKLB.

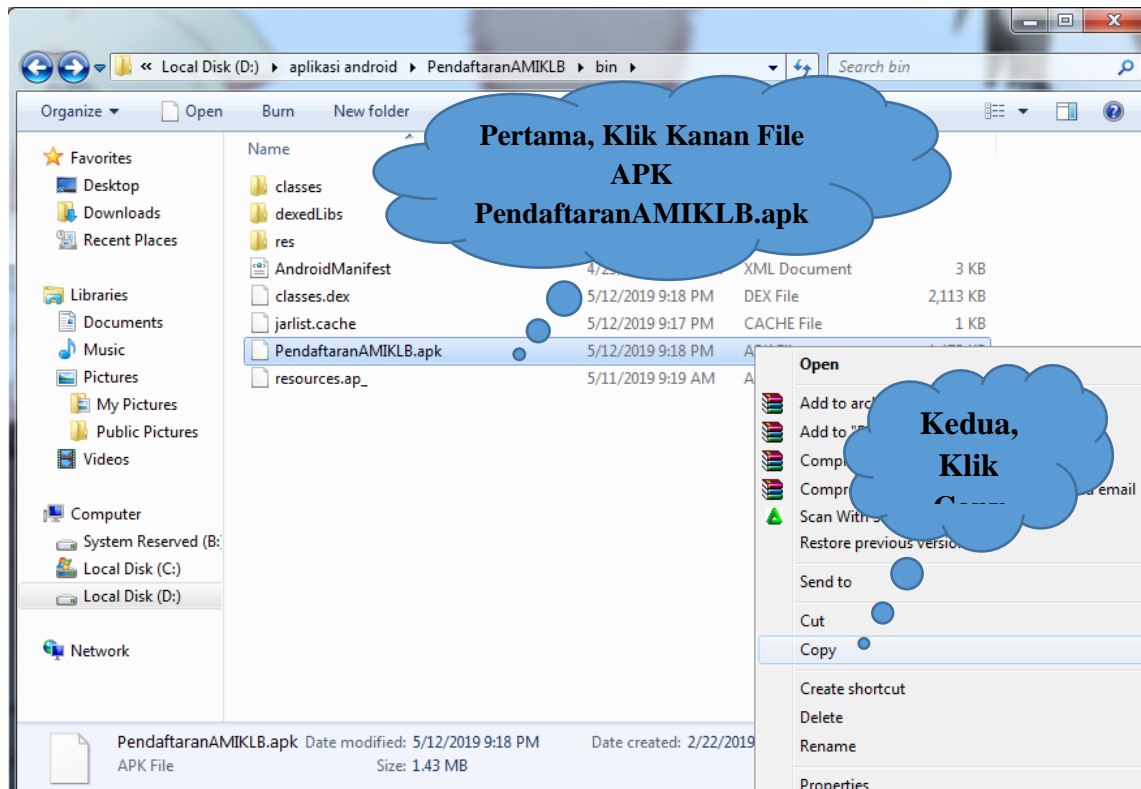
d. Folder *Bin*



Gambar 4.20 Folder *Bin*

Pada gambar 4.20 merupakan folder *bin*, disini *user* dapat mengklik folder *bin* untuk melakukan pengcopyan *file* APK aplikasi pendaftaran mahasiswa baru berbasis *android*.

e. File APK



Gambar 4.21 File APK

Pada gambar 4.21 merupakan tampilan file APK, disini *user* dapat melakukan pengcopian *file* PendaftaranAMIKLB.apk dengan cara mengklik *file* PendaftaranAMIKLB.apk, selanjutnya klik kanan *file* PendaftaranAMIKLB.apk dan pilih *copy* untuk melakukan pengcopian *file* APK aplikasi pendaftaran mahasiswa baru berbasis *android* ke media penyimpanan *Smartphone*.

2. Instalasi File Aplikasi pendaftaran AMIK Labuhanbatu di Smartphone

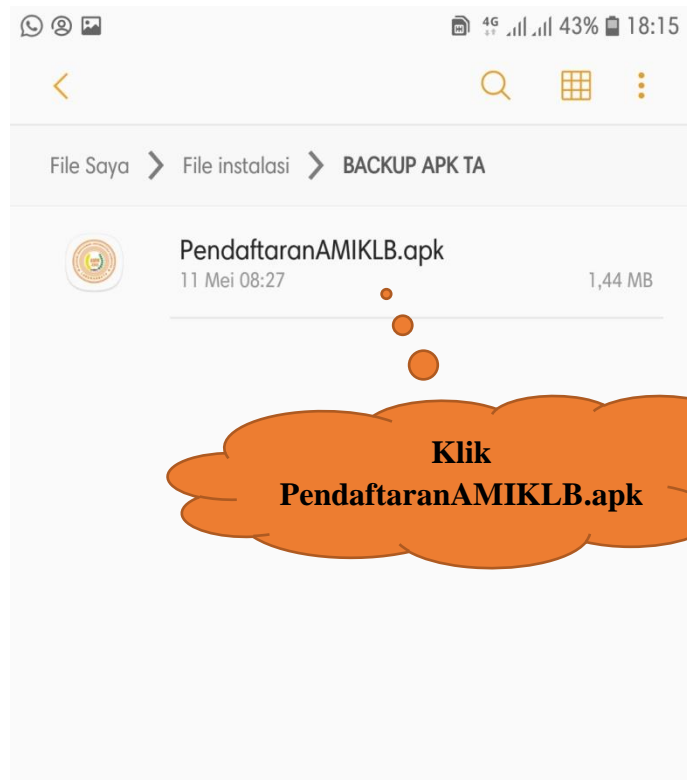
a. Tampilan Menu Aplikasi Samsung Galaxy J7 Prime



Gambar 4.22 Tampilan Menu Aplikasi Samsung Galaxy J7 Prime

Pada gambar 4.22 merupakan tampilan menu *samsung galaxy J7 prime*, disini *user* dapat mengklik folder “*file Saya*” untuk memulai penginstalan aplikasi pendaftaran mahasiswa baru berbasis *android*.

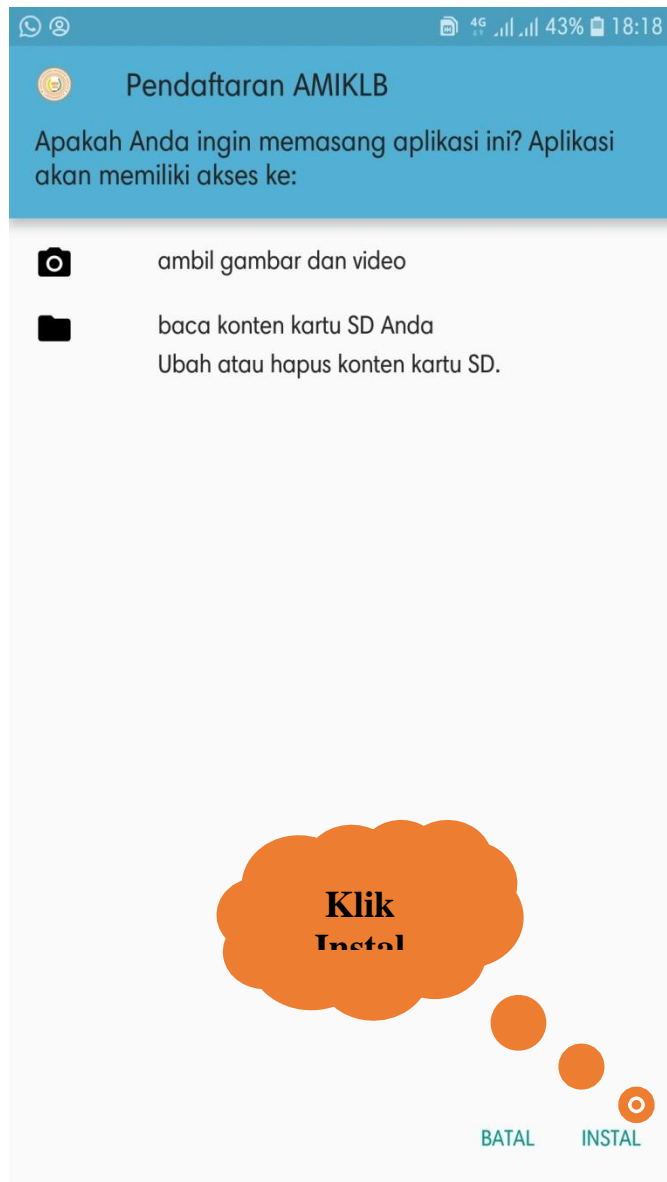
b. Tampilan Penyimpanan *File* Aplikasi Pendaftaran di *Smartphone*



Gambar 4.23 Tampilan Penyimpanan *File* Aplikasi

Pada gambar 4.23 merupakan tampilan Penyimpanan *File* aplikasi pendaftaran mahasiswa baru berbasis *android*, disini *user* dapat mengklik “PendaftaranAMIKLB.apk” untuk memulai proses penginstalan aplikasi pendaftaran mahasiswa baru berbasis *android*.

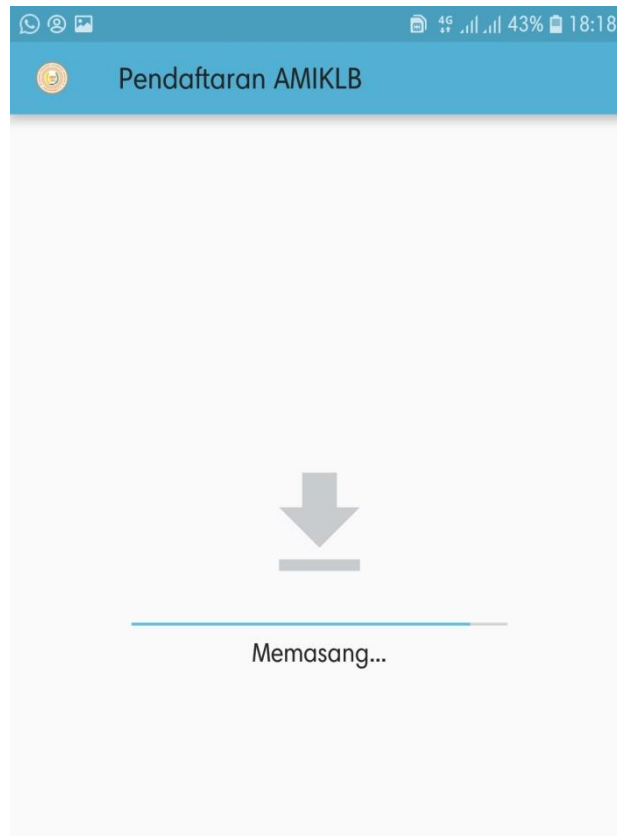
c. Tampilan Konfirmasi Penginstalan



Gambar 4.24 Tampilan Konfirmasi Penginstalan

Pada gambar 4.24 merupakan tampilan konfirmasi pengistalan, disini sistem android meminta persetujuan/konfirmasi dari *user*, jika *user* ingin melanjutkan penginstalan klik tombol "*instal*" dan jika *user* tidak ingin melanjutkan penginstalan APK klik tombol "*batal*".

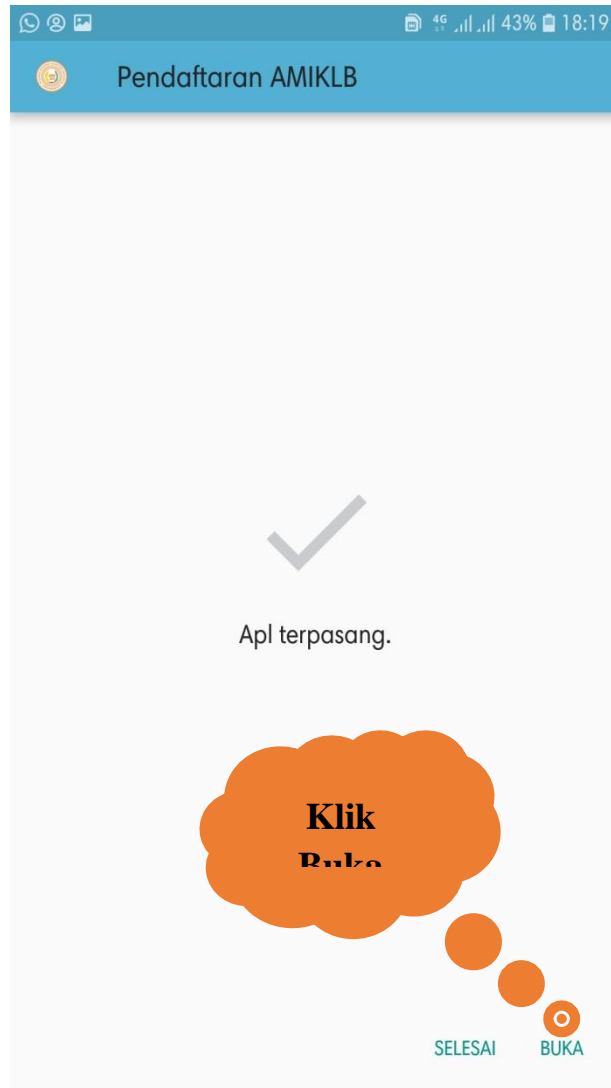
d. Tampilan *Instalasi*



Gambar 4.25 Tampilan *Instalasi*

Pada tampilan 4.25 merupakan tampilan *instalasi*, tampilan ini merupakan tampilan proses loading saat sistem sedang menginstal “PendaftaranAMIKLB.apk”.

e. Tampilan *Instal Selesai*



Gambar 4.26 Tampilan *Instal Selesai*

Pada tampilan 4.26 merupakan tampilan instal berhasil, disini proses penginstalan “PendaftaranAMIKLB.apk” telah selesai. Jika *user* ingin membuka aplikasi pendaftaran mahasiswa baru berbasis *android* klik tombol “buka”, dan jika tidak klik tombol “selesai”.

**3. Pengujian Aplikasi pendaftaran AMIK Labuhanbatu di
*Smartphone***

**a. Tampilan Icon PendaftaranAMIKLB.apk di *Samsung Galaxy J7
Prime***



Gambar 4.27 Tampilan Icon PendaftaranAMIKLB.apk

Pada gambar 4.27 merupakan tampilan icon PendaftaranAMIKLB.apk di *samsung galaxy J7 prime*, disini *user* dapat mengklik icon “PendaftaranAMIKLB.apk” untuk memulai aplikasi pendaftaran mahasiswa baru berbasis *android*.

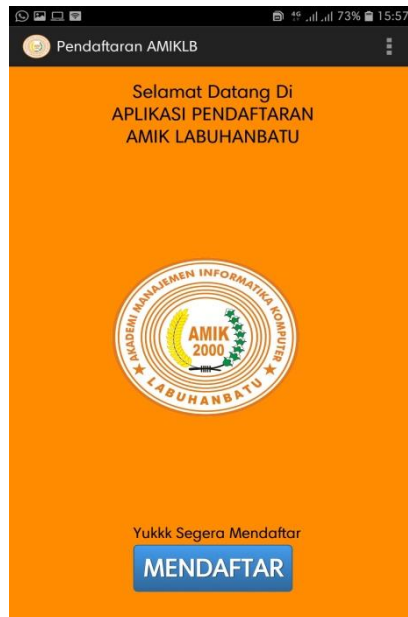
b. Tampilan *Splashscreen*



Gambar 4.28 Tampilan *Splashscreen*

Pada gambar 4.28 merupakan tampilan *splashscreen* yang ada di Aplikasi Pendaftaran Mahasiswa Baru Berbasis Android (Studi Kasus : AMIK Labuhanbatu) setelah mengklik icon “PendaftaranAMIKLB.apk“ di menu *smartphone samsung galaxy J7 prime* dan tampilan sebelum masuk ke menu utama. Di dalam *splashscreen* ini terdapat Gambar Icon dari AMIK Labuhanbatu, bacaan aplikasi pendaftaran mahasiswa baru AMIK Labuhanbatu, icon *loading* serta tulisan tunggu sebentar.

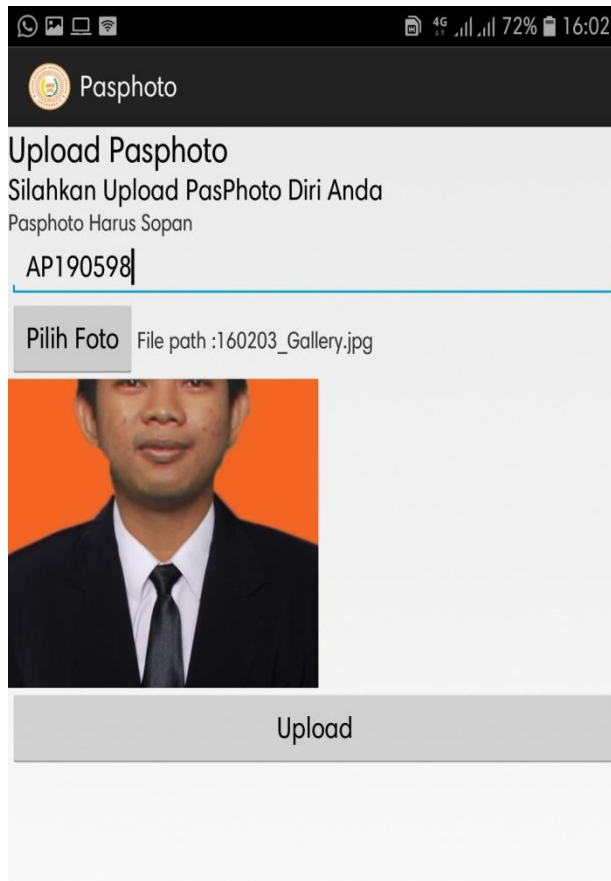
c. Tampilan Menu Utama



Gambar 4.29 Tampilan Menu Utama

Pada gambar 4.29 merupakan tampilan menu utama yang ada di Aplikasi Pendaftaran Mahasiswa Baru Berbasis Android (Studi Kasus : AMIK Labuhanbatu) Di dalam menu utama ini terdapat Gambar Icon dari AMIK Labuhanbatu, bacaan selamat datang di aplikasi pendaftaran AMIK Labuhanbatu, tulisan yukkk segera mendaftar dan tombol “MENDAFTAR”.

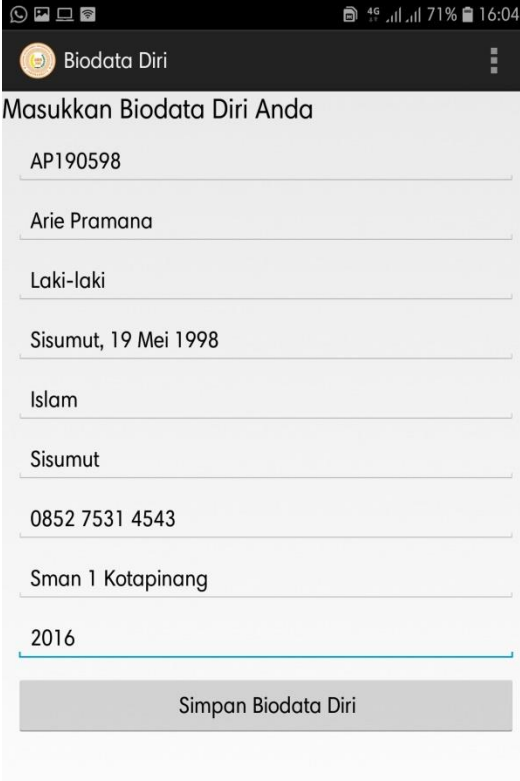
d. Tampilan Input Pasfoto



Gambar 4.30 Tampilan Input Pasfoto

Pada gambar 4.30 merupakan contoh tampilan penginputan pasfoto. Penginputan dimulai dengan mengetik Id Pendaftaran “AP190598” di *textfield*. Dengan Nama File Foto yang diupload “160203_Gallery.jpg”, Dan ada Preview Foto dari pasfoto yang akan diupload. Selanjutnya *user* mengklik tombol “Upload” dan penginputan pasfoto akan diproses dan tersimpan.

e. Tampilan Input Biodata

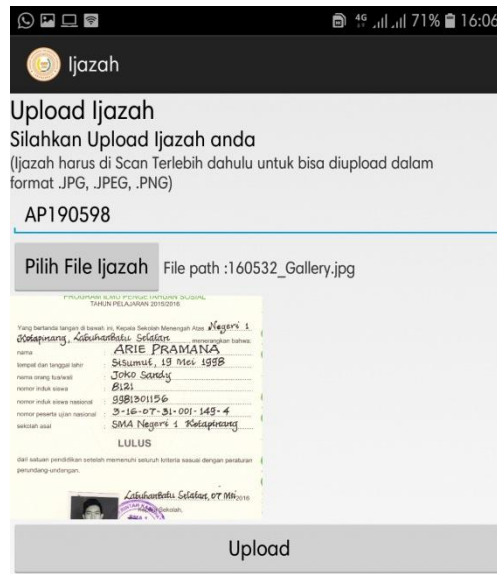


The screenshot displays a mobile application interface titled "Biodata Diri". The main heading is "Masukkan Biodata Diri Anda". Below this, there are ten text input fields, each containing a specific piece of personal information. The fields are: "AP190598", "Arie Pramana", "Laki-laki", "Sisumut, 19 Mei 1998", "Islam", "Sisumut", "0852 7531 4543", "Sman 1 Kotapinang", "2016", and a button labeled "Simpan Biodata Diri" at the bottom.

Gambar 4.31 Tampilan Input Biodata

Pada gambar 4.31 merupakan contoh tampilan penginputan biodata. Penginputan keseluruhan mengetik di *textfield*. Pada Gambar 4.10 di inputkan Id pendaftaran “AP190598”, nama lengkap “Arie Pramana”, jenis kelamin “Laki-laki”, tempat/tanggal lahir “Sisumut, 19 Mei 1998”, agama “Islam”, alamat “Sisumut” no. handphone “0852 7531 4543” asal sekolah “Sman 1 Kotapinang” dan tahun tamat “2016”. Kemudian *User* mengklik tombol “Simpan Biodata Diri” untuk menyimpan penginputan biodata.

f. Tampilan Input Ijazah



Gambar 4.32 Tampilan Input Lampiran

Pada gambar 4.32 merupakan contoh tampilan penginputan lampiran Ijazah. Penginputan dimulai dengan mengetik Id Pendaftaran “AP190598” di *textfield*. Dengan Nama File ijazah yang diupload “160532_Gallery.jpg”, Dan ada Preview Foto ijazah dari ijazah yang akan diupload. Selanjutnya *user* mengklik tombol “Upload” dan penginputan lampiran(ijazah/skhu) akan diproses dan tersimpan.

4.2.1 Pengujian Fungsional

Pengujian Fungsional ini dilakukan dengan menggunakan metode *black box*. Pengujian fungsional dilakukan pengujian sebagai *user* dan *admin*.

Tabel 4.1 Skenario Pengujian *user*

Item uji	Detail Pengujian	Jenis uji
Form Pasfoto	Input Pasfoto	Black Box
Form Biodata	Input Biodata	Black Box
Form Lampiran	Input Lampiran (Ijazah/Skhu)	Black Box

Tabel 4.2 Skenario Pengujian *admin*

Item uji	Detail Pengujian	Jenis uji
login	Verifikasi Data Login	Black Box
Data Calon Mahasiswa	Menampilkan Seluruh Data Calon Mahasiswa	Black Box

4.2.2 Kasus dan Hasil Pengujian

Berdasarkan pengujian fungsional yang telah dibuat maka dibuatlah pengujian pada sistem dengan tabel pengujian untuk setiap uji yang dilakukan pada bagian yang diisikan oleh data yang benar dan data yang salah.

Tabel 4.3 Tabel Pengujian Pafoto Benar

Kasus Dan Hasil Uji (Data Benar)	
Data yang diinputkan	Id Pendaftaran : AP190598 Pafoto : 160203_Gallery.jpg
Yang diharapkan	Jika Id Pendaftaran dan Pafoto sudah diinputkan, Aplikasi akan memproses data tersebut untuk disimpan
Pengamatan	Id pendaftaran dan pafoto dapat diinputkan, tombol Upload berfungsi, Proses penginputan pafoto dapat dilakukan.
Kesimpulan	Pafoto berhasil diinputkan

Tabel 4.4 Tabel Pengujian Pafoto Salah

Kasus Dan Hasil Uji (Data Salah)	
Data yang diinputkan	Id pendaftaran : AA0000 Pafoto :
Yang diharapkan	Dapat menampilkan pesan jika pafoto kosong
Pengamatan	Pesan kesalahan muncul no image selected.
Kesimpulan	Proses Penginputan tidak dapat dilakukan.

Tabel 4.5 Tabel Pengujian Biodata Benar

Kasus Dan Hasil Uji (Data Benar)	
Data yang diinputkan	Id Pendaftaran : AP190598 nama lengkap : Arie Pramana jenis kelamin : Laki-laki tempat/tanggal lahir : Sisumut, 19 Mei 1998 agama : Islam alamat : Sisumut no. handphone : 0852 7531 4543 asal sekolah : Sman 1 Kotapinang tahun tamat : 2016.
Yang diharapkan	Jika Id Pendaftaran, Nama Lengkap, Jenis Kelamin, Tempat/Tanggal Lahir, Agama, Alamat, No. Handphone, Asal Sekolah, Tahun Tamat sudah diinputkan, Aplikasi akan memproses data tersebut untuk disimpan
Pengamatan	Id Pendaftaran, Nama Lengkap, Jenis Kelamin, Tempat/Tanggal Lahir, Agama, Alamat, No. Handphone, Asal Sekolah, Tahun Tamat dapat diinputkan, tombol Simpan berfungsi, Proses penginputan biodata dapat dilakukan.
Kesimpulan	Biodata berhasil diinputkan

Tabel 4.6 Tabel Pengujian Biodata Salah

Kasus Dan Hasil Uji (Data Salah)	
Data yang diinputkan	Id Pendaftaran : AP190598 nama lengkap : Arie Pramana jenis kelamin : Laki-laki

	tempat/tanggal lahir : Sisumut, 19 Mei 1998 agama : alamat : no. handphone : 0852 7531 4543 asal sekolah : Sman 1 Kotapinang tahun tamat : 2016
Yang diharapkan	Dapat menampilkan pesan salah satu <i>textfield</i> kosong
Pengamatan	Pesan kesalahan muncul Gagal Mohon Diperiksa Kembali Data Anda!!!.
Kesimpulan	Proses Penginputan tidak dapat dilakukan.

Tabel 4.7 Tabel Pengujian Lampiran Benar

Kasus Dan Hasil Uji (Data Benar)	
Data yang diinputkan	Id Pendaftaran : AP190598 File ijazah : 160532_Gallery.jpg
Yang diharapkan	Jika Id Pendaftaran dan file ijazah sudah diinputkan, Aplikasi akan memproses data tersebut untuk disimpan
Pengamatan	Id pendaftaran dan file ijazah dapat diinputkan, tombol Upload berfungsi, Proses penginputan Lampiran ijazah dapat dilakukan.
Kesimpulan	Lampiran(ijazah/skhu) berhasil diinputkan

Tabel 4.8 Tabel Pengujian Lampiran Salah

Kasus Dan Hasil Uji (Data Salah)	
Data yang diinputkan	Id pendaftaran : AA0000 File ijazah :
Yang diharapkan	Dapat menampilkan pesan jika file ijazah kosong
Pengamatan	Pesan kesalahan muncul no image selected.
Kesimpulan	Proses Penginputan tidak dapat dilakukan.

Tabel 4.9 Tabel Pengujian Login Admin Benar

Kasus Dan Hasil Uji (Data Benar)	
Data yang diinputkan	Mengisi data login, contoh : Username : AMIK Labuhanbatu Password : semakinmaju
Yang diharapkan	Jika data login diinputkan dan valid, maka admin

	akan masuk kedalam sistem dengan hak akses admin.
Pengamatan	Id pendaftaran dan file ijazah dapat diinputkan, tombol Upload berfungsi, Proses penginputan Lampiran ijazah dapat dilakukan.
Kesimpulan	Diterima

Tabel 4.10 Tabel Pengujian Login Admin Salah

Kasus Dan Hasil Uji (Data Salah)	
Data yang diinputkan	Username : Arie (data salah) Password : pramana(data salah)
Yang diharapkan	Dapat menampilkan pesan jika username atau password salah
Pengamatan	Pesan kesalahan muncul Login gagal!! Periksa Kembali Username & Password!.
Kesimpulan	Diterima.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan hasil perancangan dan implementasi aplikasi pendaftaran mahasiswa baru (studi kasus : AMIK Labuhanbatu), maka dapat disimpulkan bahwa :

1. Aplikasi pendaftaran mahasiswa baru (studi kasus : AMIK Labuhanbatu) dapat mempermudah calon mahasiswa yang ingin mendaftar di AMIK Labuhanbatu disaat terkendala jarak dan waktu. Sehingga nantinya calon mahasiswa hanya tinggal memverifikasi pendaftarannya dengan mengantarkan file-file berkas asli miliknya ke AMIK Labuhanbatu.
2. Aplikasi pendaftaran mahasiswa baru (studi kasus : AMIK Labuhanbatu) dapat membantu dan mempermudah staff administrasi BAAK dalam pekerjaannya untuk mengatur dan mengelola pendaftaran mahasiswa baru.
3. Aplikasi pendaftaran mahasiswa baru (studi kasus : AMIK Labuhanbatu) dapat diimplementasikan pada sistem operasi *android* versi 2.2 sampai dengan versi 9.0.

5.2 Saran

Berikut ini merupakan beberapa saran yang dapat digunakan dalam melakukan pengembangan sistem aplikasi untuk kedepannya agar menjadi aplikasi yang lebih baik lagi, yaitu :

1. Memperbaiki input id pendaftaran, id pendaftaran cukup sekali input di layout pertama dan layout seterusnya terinput secara otomatis dengan id pendaftaran yang sama.
2. Membuat login untuk user dengan id pendaftarannya setelah selesai mendaftar.
3. Menambahkan fitur pembayaran uang administrasi pendaftaran.
4. Menambahkan fitur cetak kartu ujian setelah calon mahasiswa melakukan pendaftaran.
5. Menambahkan fitur info pendaftaran & jadwal ujian yang terhubung dengan *web* pendaftaran mahasiswa baru AMIK Labuhanbatu.

**APLIKASI PENDAFTARAN MAHASISWA BARU
BERBASIS *ANDROID*
(STUDI KASUS : AMIK LABUHANBATU)**

TUGAS AKHIR

Untuk Memenuhi Persyaratan Memperoleh Gelar Ahli Madya Pada Program
Studi Manajemen Informatika Fakultas Sains dan Teknologi
Universitas Labuhanbatu



Oleh:

ARIE PRAMANA

16.051.00.006

**PROGRAM STUDI MANAJEMEN INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS LABUHANBATU
RANTAUPRAPAT**

2019

xcvi

DAFTAR GAMBAR

Gambar 2.1	: Alur Suatu Sistem	6
Gambar 2.2	: Eclipse	21
Gambar 2.3	: Tampilan AVD	23
Gambar 2.4	: XAMPP Logo	24
Gambar 2.5	: AMIK Labuhan Batu	36
Gambar 2.6	: Struktur Organisasi Institusi AMIK Labuhanbatu	38
Gambar 3.1	: Use Case Diagram	55
Gambar 3.2	: Activity Diagram	56
Gambar 3.3	: Sequence Diagram	58
Gambar 3.4	: Rancangan Form Pasfoto	59
Gambar 3.5	: Rancangan Form Biodata	60
Gambar 3.6	: Rancangan Form Lampiran	61
Gambar 3.7	: Relasi Tabel	67
Gambar 3.8	: Component Diagram User	72
Gambar 3.9	: Component Diagram Admin	73
Gambar 4.1	: Tabel Admin	77
Gambar 4.2	: Tabel Biodata	77
Gambar 4.3	: Tabel Ijazah	78
Gambar 4.4	: Tabel Pasfoto	78
Gambar 4.5	: Splashscreen	79
Gambar 4.6	: Menu Utama	80
Gambar 4.7	: Form Pasfoto	81
Gambar 4.8	: Contoh Input Pasfoto	82
Gambar 4.9	: Form Biodata	83
Gambar 4.10	: Contoh Input Biodata	84
Gambar 4.11	: Form Lampiran	85
Gambar 4.12	: Contoh Input Lampiran	86
Gambar 4.13	: Tampilan Login Admin	87
Gambar 4.14	: Input Login Admin	87
Gambar 4.15	: Tampilan Home	88

Gambar 4.16 : Tampilan Data Calon Mahasiswa	89
Gambar 4.17 : Local Disk D	90
Gambar 4.18 : Folder Aplikasi Android	91
Gambar 4.19 : Folder Pendaftaran AMIKLB	92
Gambar 4.20 : Folder Bin	92
Gambar 4.21 : File APK	93
Gambar 4.22 : Tampilan Menu Aplikasi Samsung Galaxy J7 Prime	94
Gambar 4.23 : Tampilan Penyimpanan File Aplikasi	95
Gambar 4.24 : Tampilan Konfirmasi Penginstalan	96
Gambar 4.25 : Tampilan Instalasi	97
Gambar 4.26 : Tampilan Instal Selesai	98
Gambar 4.27 : Tampilan Icon PendaftaranAMIKLB.apk	99
Gambar 4.28 : Tampilan Splashscreen	100
Gambar 4.29 : Tampilan Menu Utama	101
Gambar 4.30 : Tampilan Input Pasfoto	102
Gambar 4.31 : Tampilan Input Biodata	103
Gambar 4.32 : Tampilan Input Lampiran	104

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
LEMBAR PENGESAHAN	iii
SURAT PERNYATAAN	iv
ABSTRAK	v
ABSTRACT	vi
KATA PENGANTAR	vii
DAFTAR ISI	ix
DAFTAR TABEL	xii
DAFTAR GAMBAR	xiii
DAFTAR LAMPIRAN	xv

BAB I	: PENDAHULUAN	1
1.1	Latar Belakang Masalah	1
1.2	Perumusan Masalah	2
1.3	Batasan Masalah	3
1.4	Tujuan Penelitian	3
1.5	Sistematika Penulisan	4
BAB II	: LANDASAN TEORI	6
2.1	Definisi Sistem	6
2.1.1	Karakteristik Sistem	7
2.1.2	Klasifikasi Sistem	9
2.2	Aplikasi	10
2.3	Konsep Dasar Pendaftaran Mahasiswa Baru	11
2.3.1	Pendaftaran	11
2.3.2	Mahasiswa	11
2.3.3	Penerimaan Mahasiswa Baru	12
2.4	Android	13
2.4.1	Sejarah Android	13
2.4.2	Perkembangan Android	15
2.4.3	Android SDK (<i>Software Development Kit</i>)	19
2.4.4	<i>Android Development Tools</i> (ADT)	19
2.5	Eclipse	20
2.6	<i>Java Development Kit</i> (JDK)	21
2.7	<i>Android Virtual Device</i> (AVD)	22
2.8	XAMPP	23
2.9	PHP (<i>Hypertext Preprocessor</i>)	24
2.10	Mysql	26
2.11	<i>Unified modelling language</i> (UML)	27
2.11.1	Use Case Diagram	28
2.11.2	Activity Diagram	29
2.11.3	Sequence Diagram	31
2.11.4	Component Diagram	32
2.12	AMIK Labuhanbatu	33

2.12.1	Visi AMIK Labuhanbatu	34
2.12.2	Misi AMIK Labuhanbatu	35
2.12.3	Tujuan AMIK Labuhanbatu	35
2.12.4	Struktur Organisasi AMIK Labuhanbatu	36
2.12.5	Tugas Pokok dan Fungsi Struktur AMIK Labuhanbatu	38
BAB III	: METODOLOGI PENELITIAN	53
3.1	Metode Pengumpulan Data	53
3.2	Metode Perancangan Sistem	54
3.2.1	Use Case Diagram	55
3.2.2	Activity Diagram	56
3.2.3	Sequence Diagram	57
3.2.4	Rancangan Masukan (Input)	57
3.2.5	Rancangan Proses (Process)	62
3.2.6	Rancangan Keluaran (Output)	62
3.2.7	Rancangan Basis Data	63
3.2.7.1	Normalisasi	64
3.2.7.2	Relasi Tabel	67
3.2.7.3	Desain File/Tabel	68
3.2.8	Rancangan Interface	72
BAB IV	: HASIL DAN PEMBAHASAN	74
4.1	Implementasi	74
4.1.1	Perangkat Keras	74
4.1.2	Perangkat Lunak	75
4.1.3	Implementasi Database	76
4.1.4	Implementasi Tampilan	78
4.1.4.1	Tampilan User	79
4.1.4.2	Tampilan Admin	86
4.2	Pengujian	90
4.2.2	Kasus dan Hasil Pengujian	105
BAB V	: PENUTUP	110
5.1	Kesimpulan	110

5.2 Saran	111
DAFTAR PUSTAKA	112
LAMPIRAN	L-1
1. DAFTAR RIWAYAT HIDUP	L-1
2. LISTING PROGRAM	L-2

DAFTAR LAMPIRAN

Lampiran 1 : Daftar Riwayat Hidup	L-1
Lampiran 2 : Listing Program	L-2

DAFTAR PUSTAKA

- [1] E. N. Jannah, D. K. Bayturrohman, and E. Kurniawan, “Pengembangan Aplikasi Penerimaan Mahasiswa Baru Berbasis Android Dilengkapi dengan Fitur Push Notification,” *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 6, no. 4, pp. 410–415, 2017.
- [2] K. Yasdomi *et al.*, “PEMANFAATAN TEKNOLOGI SMARTPHONE UNTUK PROSES PENERIMAAN MAHASISWA BARU SECARA ONLINE,” pp. 1–12.
- [3] S. Eviani, S. D. Rizki, and M. Pratiwi, “Sistem Informasi Penerimaan Siswa Baru Berbasis Web Pada Smpn 34 Kabupaten Tebo,” *J. Teknol.*, no. 1, pp. 13–19, 2016.
- [4] K. Ramanda, “PENERAPAN SISTEM MANAJEMEN OPERASIONAL PELAYANAN PEMESANAN MENU MAKANAN DENGAN WAITING LINE METHOD,” *J. Pilar Nusa Mandiri*, vol. XII, no. 2, pp. 182–189, 2016.
- [5] D. Akhiyar, “SISTEM PAKAR UNTUK MENDIAGNOSA PENYAKIT THT (TELINGA HIDUNG TENGGOROKAN) MENGGUNAKAN METODE FORWARD CHAINING DENGAN BAHASA PEMROGRAMAN PHP DAN MYSQL di RS M . Ibnu Sina Padang,” *J. Teknol.*, vol. 7, no. 2, pp. 248–258, 2017.
- [6] D. Polanda *et al.*, “PERANCANGAN KUIS BERBASIS ANDROID SEBAGAI MEDIA PEMBELAJARAN DI SMK SWASTA AL-AZIS,” vol. 5, no. 2, pp. 1–6, 2017.
- [7] A. R. R. Hasan Abdurahman, “Perancangan Aplikasi E-Canteen Berbasis

- Android Dengan Menggunakan Metode Object Oriented Analysis & Design (OOAD),” *J. Penelit. Komun. dan Opini Publik*, vol. 20, no. 1, pp. 83–92, 2014.
- [8] L. Magdalena and A. Rachman, “Aplikasi Pendaftaran Siswa Baru Dengan Sistem Seleksi Menggunakan Metode Simple Additive Weighting (Saw) Pada Smk Miftahul Huda Ciwaringin,” *J. Digit*, vol. 7, no. 1, pp. 38–49, 2017.
- [9] J. O. Papilaya and N. Huliselan, “Identifikasi Gaya Belajar Mahasiswa,” *J. Psikol. Undip*, vol. 15, no. 1, p. 56, 2017.
- [10] R. M. Zaef, N. C. Herbaviana, and A. Chusyairi, “Sistem Informasi Penerimaan Peserta Didik Baru Berbasis Android Menggunakan Metode Agile,” in *Konferensi Nasional Sistem Informasi*, 2018, pp. 226–231.
- [11] I. Purnama, “Sistem Informasi Kursus IBAY Komputer Berbasis Web dan Mobile Android,” *Riau J. Comput. Sci.*, vol. 05, no. 01, pp. 23–31, 2019.
- [12] N. S. P. S. Harni Kusniyati, “Aplikasi Edukasi Budaya Toba Samosir Berbasis Android,” *Tek. Inform.*, vol. 9, no. 1, pp. 9–16, 2016.
- [13] S. Andriani, “Aplikasi Akademik Online Berbasis Mobile Android Pada UNIVERSITAS TAMA JAGAKARSA,” *J. Sains dan Teknol. Utama, Vol. XI, Nomor 1, April 2016*, vol. XI, no. 152, p. 12, 2016.
- [14] S. A. Pratiwi, I. M. Sukarsa, and I. K. A. Purnawan, “Rancang Bangun Aplikasi Sistem Pemesan Bunga Berbasis Android,” *J. Ilm. Merpati (Menara Penelit. Akad. Teknol. Informasi)*, vol. 2, no. 2, pp. 205–214, 2016.
- [15] A. Sinsuw and X. Najoan, “Prototipe Aplikasi Sistem Informasi Akademik

- Pada Perangkat Android,” *Tek. Elektro dan Komput.*, vol. 2, no. 5, pp. 1–10, 2013.
- [16] R. V Palit, Y. D. Y. Rindengan, and A. S. M. Lumenta, “Rancangan Sistem Informasi Keuangan Berbasis Web Di Jemaat GMIM Bukit Moria Malalayang,” *E-Journal Tek. Elektro dan Komput. vol.*, vol. 4, no. 7, pp. 1–7, 2015.
- [17] A. Firman, H. Wowor, and X. Najoan, “Sistem Informasi Perpustakaan Online Berbasis Web Application,” *J. Sist. Inf. Bisnis*, vol. 1, no. 2, pp. 66–77, 2016.
- [18] A. Hendini, “PEMODELAN UML SISTEM INFORMASI MONITORING PENJUALAN DAN STOK BARANG (STUDI KASUS: DISTRO ZHEZHA PONTIANAK),” *J. KHATULISTIWA Inform.*, vol. IV, no. 2, pp. 107–116, 2016.
- [19] H. Tohari, *ANALISIS SERTA PERANCANGAN SISTEM INFORMASI MELALUI PENDEKATAN UML*. Yogyakarta: Andi, 2014.

LAMPIRAN

Lampiran 1.

DAFTAR RIWAYAT HIDUP

Saya yang bertanda tangan dibawah ini :

Nama : Arie Pramana
Tempat, Tanggal Lahir : Sisumut, 19 Mei 1998
Jenis kelamin : Laki – laki
Agama : Islam
Status : Belum Menikah
Alamat : Sisumut
No.HP : 085275314543

PENDIDIKAN FORMAL

1. SD Negeri 115495 Sisumut Berijazah
2. SMP Negeri 2 Kotapinang Berijazah
3. SMA Negeri 1 Kotapinang Berijazah
4. AMIK Labuhan Batu 2016 – 2019

Demikian daftar riwayat hidup ini saya perbuat dengan sebenarnya.

Rantauprapat, 31 Juli 2019

Saya yang membuat

ARIE PRAMANA

DAFTAR TABEL

Tabel 2.1 : Use Case Diagram	28
Tabel 2.2 : Activity Diagram	29
Tabel 2.3 : Sequence Diagram	31
Tabel 2.4 : Component Diagram	32
Tabel 3.1 : Rancangan Output Form Tabel Data Calon Mahasiswa Baru	63
Tabel 3.2 : Tabel Database Mahasiswa Baru	64
Tabel 3.3 : Tabel Bentuk Tidak Normal	65
Tabel 3.4 : Tabel Bentuk Normal Pertama	65
Tabel 3.5 : Tabel Biodata	66
Tabel 3.6 : Tabel Pasfoto	66
Tabel 3.7 : Tabel Ijazah	67
Tabel 3.8 : Struktur Tabel Admin	69
Tabel 3.9 : Struktur Tabel Biodata	70
Tabel 3.10 : Struktur Tabel Ijazah	71
Tabel 3.11 : Struktur Tabel Pasfoto	71
Tabel 4.1 : Skenario Pengujian User	105
Tabel 4.2 : Skenario Pengujian Admin	105
Tabel 4.3 : Tabel Pengujian Pasfoto Benar	106
Tabel 4.4 : Tabel Pengujian Pasfoto Salah	106
Tabel 4.5 : Tabel Pengujian Biodata Benar	107
Tabel 4.6 : Tabel Pengujian Biodata Salah	108
Tabel 4.7 : Tabel Pengujian Lampiran Benar	108
Tabel 4.8 : Tabel Pengujian Lampiran Salah	109
Tabel 4.9 : Tabel Pengujian Login Admin Benar	109
Tabel 4.10 : Tabel Pengujian Login Admin Salah	109

KATA PENGANTAR

Puji syukur penulis ucapkan kepada Allah Subhanahu wa Ta'ala, atas segala nikmat iman, kehidupan, dan kesempatan mengenyam ilmu, sehingga penulis dapat menyelesaikan Tugas Akhir ini yang mana disusun sebagai salah satu persyaratan mencapai status Diploma III Manajemen Informatika di AMIK Labuhan Batu.

Pada kesempatan ini penulis mengucapkan terima kasih kepada orang tua kami Bapak Joko dan Ibu Siti Khadijah serta seluruh keluarga atas cinta dan doa terus menerus kepada kami agar kami dapat menyelesaikan Tugas Akhir ini. Ucapan terima kasih dan penghargaan sebesar-besarnya juga kami haturkan kepada:

1. Bapak Dr. H. Amarullah Nasution, SE.,MBA, Selaku Ketua Yayasan Universitas Labuhanbatu.
2. Ibu Deci Irmayani, S.Kom.,M.Kom, Selaku Direktur AMIK Labuhanbatu.
3. Ibu Marnis Nasution, S.Kom.,M.Kom, Selaku Ketua Program Studi Manajemen Informatika dan Dosen Pembimbing I yang telah membimbing penulis dari awal penulisan hingga akhir.
4. Bapak Iwan Purnama, S.Kom.,M.Kom Selaku Ketua Program Studi Teknologi Informasi dan Dosen Pembimbing II yang telah membimbing penulis dari awal penulisan hingga akhir.
5. Bapak/Ibu Dosen serta staff AMIK Labuhanbatu yang telah memberikan ilmu pengetahuan kepada penulis.

6. Seluruh teman-teman seperjuangan di kelas MI-A atas ilmu, pengalaman, dan persahabatan yang diberikan kepada kami.

Tidak ada gading yang tak retak, tidak ada karya yang sempurna, karena kesempurnaan hanya milik Allah semata. Akhir kata, semoga Tugas Akhir ini menjadi amal bagi penulis, amin ya rabbala'lamin.

Rantauprapat, 31 Juli 2019
Penulis,

ARIE PRAMANA
16.051.00.006

LEMBARAN PENGESAHAN NASKAH TUGAS AKHIR

JUDUL TUGAS AKHIR : APLIKASI PENDAFTARAN MAHASISWA BARU
BERBASIS ANDROID (STUDI KASUS : AMIK
LABUHANBATU)

NAMA : ARIE PRAMANA
NPM : 16.051.00.006
PRODI : MANAJEMEN INFORMATIKA

Telah Diuji Dan Dinyatakan Lulus Dalam Ujian Ahli Madya
Pada Tanggal 31 Juli 2019

TIM PENGUJI

Tanda Tangan

Penguji I (Ketua)

Nama : Deci Irmayani,S.Kom,M.Kom
NIDN : 0127058602

Penguji II (Anggota)

Nama : Marnis Nasution,S.Kom,M.Kom
NIDN : 0130039001

Penguji II (Anggota)

Nama : Iwan Purnama,S.Kom,M.Kom
NIDN : 0112029202

Rantauprapat, 31 juli 2019

**Dekan,
Fakultas Sains dan Teknologi**

**Ka. Program Studi,
Manajemen Informatika**

(Novilda Elizabeth Mustamu, S.Pt, M.Si)
NIDN. 0112117802

(Marnis Nasution,S.Kom,M.Kom)
NIDN. 0130039001

LEMBARAN PENGESAHAN/PERSETUJUAN SKRIPSI

JUDUL TUGAS AKHIR : APLIKASI PENDAFTARAN MAHASISWA BARU
BERBASIS ANDROID (STUDI KASUS : AMIK
LABUHANBATU)

NAMA : ARIE PRAMANA

NPM : 16.051.00.006

PRODI : MANAJEMEN INFORMATIKA

Disetujui Pada Tanggal : 31 Juli 2019

Pembimbing I,

Pembimbing II

(Marnis Nasution, S.Kom, M.Kom)
NIDN. 0130039001

(Iwan Purnama, S.Kom, M.Kom)
NIDN. 0112029202

Lampiran 2. LISTING PROGRAM

ANDROID - > USER

File Php

tambah.php

```
<?php
$conn = mysql_connect("localhost","root","");
mysql_select_db("mahasiswabaruu");
$Id      = $_POST['id_pendaftaran'];
>Nama    = $_POST['nama'];
>Jenis_kelamin = $_POST['jenis_kelamin'];
>Ttl     = $_POST['ttl'];
>Alamat  = $_POST['alamat'];
>Agama  = $_POST['agama'];
>Hp     = $_POST['hp'];
>Asal   = $_POST['asal'];
>Tahun  = $_POST['tahun'];

header('Content-Type: text/xml');
$query   = "insert into biodata (id_pendaftaran,
nama, jenis_kelamin, ttl, alamat, agama, hp, asal, tahun) values ('$Id', '$N
ama', '$Jenis_kelamin', '$Ttl', '$Alamat', '$Agama', '$Hp', '$Asal',
'$Tahun')";
$hasil = mysql_query($query);
if($hasil)
{
$response["success"] = "1";
$response["message"] = "Data sukses diinput";
echo json_encode($response);
} else { $response["success"] = "0";
$response["message"] = "Maaf , terjadi kesalahan";
// echoing JSON response
echo json_encode($response);
}
?>
```

Pasfoto - > simpan.php

```
<?php
$f=mysql_connect('localhost','root','');
mysql_select_db('mahasiswabaruu',$f);
$response = array();
if (isset($_POST['id_pendaftaran']) && isset($_POST['foto'])) {
$name = $_POST['id_pendaftaran'];
$img = $_POST['foto'];
$result = mysql_query("INSERT INTO pasfoto(id_pendaftaran, foto)
VALUES('$name', '$img')");
// cek data udah masuk belum
if ($result) {
// kalo sukses
$response["success"] = 1;
$response["message"] = "Tambah berhasil";
// echoing JSON response
echo json_encode($response);
} else {
// fkalo gagal
$response["success"] = 0;
$response["message"] = "Sistem mendeteksi kesalahan,
silahkan coba lagi";
```

```

        // echoing JSON response
        echo json_encode($response); }
} else {
    $response["success"] = 0;
    $response["message"] = "Silahkan lengkapi aksi sebelum memulai
permintaan anda";
    // echoing JSON response
    echo json_encode($response);
}
?>

```

Pasfoto - > upload.php

```

<?php
echo '<pre>';
print_r($_FILES);
echo '</pre>';
echo '<pre>';
print_r($_POST);
echo '</pre>';
$file_path="img/";
$file_path=$file_path.basename($_FILES['file']['name']);
if (move_uploaded_file($_FILES['file']['tmp_name'],$file_path)){
echo "file save success";
} else {
echo "failed to save file";
}?>

```

Ijazah - > simpan.php

```

<?php
$f=mysql_connect('localhost','root','');
mysql_select_db('mahasiswabaruar',$f);
$response = array();
if(isset($_POST['id_pendaftaran'])&&isset($_POST['file_ijazah']))
{
    $name = $_POST['id_pendaftaran'];
    $img = $_POST['file_ijazah'];
    $result = mysql_query("INSERT INTO ijazah(id_pendaftaran,
file_ijazah) VALUES('$name','$img')");
    // cek data udah masuk belum
    if ($result) {
        // kalo sukses
        $response["success"] = 1;
        $response["message"] = "Tambah berhasil";
        // echoing JSON response
        echo json_encode($response);
    } else {
        // kalo gagal
        $response["success"] = 0;
        $response["message"] = "Sistem mendeteksi kesalahan,
silahkan coba lagi";
        // echoing JSON response
        echo json_encode($response); }
} else {
    $response["success"] = 0;
    $response["message"] = "Silahkan lengkapi aksi sebelum memulai
permintaan anda";

    // echoing JSON response
    echo json_encode($response);
}?>

```

Ijazah - > upload.php

```
<?php
echo '<pre>';
print_r($_FILES);
echo '</pre>';
echo '<pre>';
print_r($_POST);
echo '</pre>';
$file_path="img/";
$file_path=$file_path.basename($_FILES['file']['name']);
if (move_uploaded_file($_FILES['file']['tmp_name'],$file_path)){
    echo "file save success";
} else {
    echo "failed to save file";
} ?>
```

Project Android

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.ujicoba.pendaftaranamiklb"
    android:versionCode="1"
    android:versionName="1.0" >
    <uses-sdk
        android:minSdkVersion="8"
        android:targetSdkVersion="17" />
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.CAMERA" />
    <uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <application
        android:allowBackup="true"
        android:icon="@drawable/ic_launcher"
        android:label="@string/app_name"
        android:theme="@style/AppTheme" >
        <activity
android:name="com.ujicoba.pendaftaranamiklb.Splashscreen"
            android:label="@string/app_name" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category
android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
android:name="com.ujicoba.pendaftaranamiklb.Pendaftar"
            android:label="@string/title_activity_pendaftar" >
        </activity>
        <activity
android:name="com.ujicoba.pendaftaranamiklb.CalonMahasiswa"
            android:label="@string/title_activity_calon_mahasiswa" >
        </activity>
        <activity
            android:name="com.ujicoba.pendaftaranamiklb.Splash"
            android:label="@string/title_activity_splash" >
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
```

```

        <category
android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
    <activity
android:name="com.ujicoba.pendaftaranamiklb.MainActivity"
    android:label="@string/title_activity_splashscreen" >
    </activity>
    <activity
    android:name="com.ujicoba.pendaftaranamiklb.Pasfoto"
    android:label="@string/title_activity_pasfoto" >
    </activity>
    <activity
    android:name="com.ujicoba.pendaftaranamiklb.Pasphoto"
    android:label="@string/title_activity_pasphoto" >
    </activity>
    <activity
    android:name="com.ujicoba.pendaftaranamiklb.Ijazah"
    android:label="@string/title_activity_ijazah" >
    </activity>
    <activity
    android:name="com.ujicoba.pendaftaranamiklb.Selesai"
    android:label="@string/title_activity_selesai" >
    </activity>
    <activity
    android:name="com.ujicoba.pendaftaranamiklb.Pedoman"
    android:label="@string/title_activity_pedoman" >
    </activity>
</application>
</manifest>

```

CustomMultiPartEntity.java

```

package com.ujicoba.pendaftaranamiklb;
import java.io.FilterOutputStream;
import java.io.IOException;
import java.io.OutputStream;
import java.nio.charset.Charset;
import org.apache.http.entity.InputStreamEntity;
import org.apache.http.entity.StringEntity;
import org.apache.http.entity.mime.HttpMultipartMode;
import org.apache.http.entity.mime.MultipartEntity;

public class CustomMultiPartEntity extends MultipartEntity {
    private UploadProgressListene uploadProgressListener;
    public CustomMultiPartEntity() {
        super(); }
    public CustomMultiPartEntity(final HttpMultipartMode mode) {
        super(mode); }
    public CustomMultiPartEntity(HttpMultipartMode mode, final String
boundary,final Charset charset) {
        super(mode, boundary, charset); }
    @Override
    public void writeTo(final OutputStream outstream) throws
IOException {
        super.writeTo(new CountingOutputStream(outstream,
            this.uploadProgressListener)); }
    public UploadProgressListene getUploadProgressListener() {
        return uploadProgressListener; }
    public void setUploadProgressListener(

```

```

        UploadProgressListene uploadProgressListener) {
    this.uploadProgressListener = uploadProgressListener; }
    public static class CountingOutputStream extends
FilterOutputStream {
        private final UploadProgressListene uploadProgressListener;
        private long transferred;
        public CountingOutputStream(final OutputStream out,
            final UploadProgressListene uploadProgressListener) {
            super(out);
            this.uploadProgressListener = uploadProgressListener;
            this.transferred = 0; }
        @Override
    public void write(byte[] b, int off, int len) throws IOException {
        out.write(b, off, len);
        this.transferred += len;
        if (this.uploadProgressListener != null) {
            this.uploadProgressListener.transferred(this.transferred);}
        @Override
    public void write(int b) throws IOException {
        out.write(b);
        this.transferred++;
        if (this.uploadProgressListener != null) {
            this.uploadProgressListener.transferred(this.transferred);}}
    public void addPart(String string, InputStreamEntity
inputStreamBody) {
        // TODO Auto-generated method stub}
    public void addPart(String string, StringEntity dataString) {
        // TODO Auto-generated method stub}}

```

UploadProgressListene.java

```

package com.ujicoba.pendaftarannamiklb;
/**
 * Upload Listener
 *
 */
public interface UploadProgressListene {
    /**
     * This method updated how much data size uploaded to server
     *
     * @param num
     */
    void transferred(long num);
}

```

JSONParser.java

```

package com.ujicoba.pendaftarannamiklb;
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.io.UnsupportedEncodingException;
import java.util.List;
import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;
import org.apache.http.client.ClientProtocolException;
import org.apache.http.client.entity.UrlEncodedFormEntity;
import org.apache.http.client.methods.HttpGet;

```

```

import org.apache.http.client.methods.HttpPost;
import org.apache.http.client.utils.URLEncodedUtils;
import org.apache.http.impl.client.DefaultHttpClient;
import org.json.JSONException;
import org.json.JSONObject;
import android.util.Log;

public class JSONParser {
    static InputStream is = null;
    static JSONObject jsonObj = null;
    static String json = "";
    // constructor
    public JSONParser() { }
    public JSONObject getJSONFromUrl(String url) {
        try { DefaultHttpClient httpClient = new
DefaultHttpClient();
        HttpPost httpPost = new HttpPost(url);
        HttpResponse httpResponse = httpClient.execute(httpPost);
        HttpEntity httpEntity = httpResponse.getEntity();
        is = httpEntity.getContent();
    } catch (UnsupportedEncodingException e) {
        e.printStackTrace();
    } catch (ClientProtocolException e) {
        e.printStackTrace();
    } catch (IOException e) {
        e.printStackTrace();
    }
    try { BufferedReader reader = new BufferedReader(new
InputStreamReader(is, "iso-8859-1"), 8);
        StringBuilder sb = new StringBuilder();
        String line = null;
        while ((line = reader.readLine()) != null) {
            sb.append(line + "\n");
        }
        is.close();
        json = sb.toString();
    } catch (Exception e) {
        Log.e("Buffer Error", "Error converting result " +
e.toString());
    }
    try { jsonObj = new JSONObject(json);
    } catch (JSONException e) {
        Log.e("JSON Parser", "Error parsing data " + e.toString());
    }
    return jsonObj; }
    public JSONObject makeHttpRequest(String url, String method,
List<NameValuePair> params) {
        try {if (method == "POST") {
            DefaultHttpClient httpClient = new DefaultHttpClient();
            HttpPost httpPost = new HttpPost(url);
            httpPost.setEntity(new UrlEncodedFormEntity(params));
            HttpResponse httpResponse = httpClient.execute(httpPost);
            HttpEntity httpEntity = httpResponse.getEntity();
            is = httpEntity.getContent();
        } else if (method == "GET") {
            DefaultHttpClient httpClient = new DefaultHttpClient();
            String paramString = URLEncodedUtils.format(params, "utf-8");
            url += "?" + paramString;
            HttpGet httpGet = new HttpGet(url);

            HttpResponse httpResponse = httpClient.execute(httpGet);
            HttpEntity httpEntity = httpResponse.getEntity();
            is = httpEntity.getContent(); }

```



```

        } catch (UnsupportedEncodingException e) {
            e.printStackTrace();
        } catch (ClientProtocolException e) {
            e.printStackTrace();
        } catch (IOException e) {
            e.printStackTrace();
        }
        try {
            BufferedReader reader = new BufferedReader(
new InputStreamReader(is, "iso-8859-1"), 8);
            StringBuilder sb = new StringBuilder();
            String line = null;
            while ((line = reader.readLine()) != null) {
                sb.append(line + "\n");
            }
            is.close();
            json = sb.toString();
        } catch (Exception e) {
            Log.e("Buffer Error", "Error converting result " +
e.toString());
        }
        try {
            JSONObject jsonObj = new JSONObject(json);
        } catch (JSONException e) {
            Log.e("JSON Parser", "Error parsing data " +
e.toString());
        }
        return jsonObj;
    }
}

```

Splashscreen.java

```

package com.ujicoba.pendaftarannamiklb;
import android.os.Bundle;
import android.os.Handler;
import android.app.Activity;
import android.content.Intent;
import android.view.Menu;

public class Splashscreen extends Activity {
    private final int time = 4000;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_splashscreen);
        new Handler().postDelayed(new Runnable() {
            @Override
            public void run() {
                Intent splashIntent = null;
                splashIntent = new Intent(Splashscreen.this, MainActivity.class);
                Splashscreen.this.startActivity(splashIntent);
                Splashscreen.this.finish();
            },time);
        }
    }
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.splashscreen, menu);
        return true;
    }
}

```

MainActivity.java

```

package com.ujicoba.pendaftarannamiklb;
import android.os.Bundle;
import android.app.Activity;
import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.view.Menu;

```

```

import android.view.View;

public class MainActivity extends Activity {
    @Override
        protected void onCreate(Bundle savedInstanceState) {
            super.onCreate(savedInstanceState);
            setContentView(R.layout.activity_main);}
    public void move(View view){
    Intent intent = new Intent(MainActivity.this, Pedoman.class);
        startActivity(intent); }
    public void onBackPressed(){
        AlertDialog.Builder builder = new AlertDialog.Builder(this);
        //pesan Keluar
        builder.setMessage("Keluar dari Aplikasi ?")
        .setCancelable(false)
        //Button Keluar
        .setPositiveButton("Ya", new DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog,int id){
                finish();
                moveTaskToBack(true);
                System.exit(0); } })
        //Button Batal
        .setNegativeButton("Tidak", new DialogInterface.OnClickListener()
        {public void onClick(DialogInterface dialog,int id){
            dialog.cancel(); } });
        AlertDialog alert = builder.create();
        alert.show(); }
    @Override
        public boolean onCreateOptionsMenu(Menu menu) {
            // Inflate the menu; this adds items to the action bar
            if it is present.
                getMenuInflater().inflate(R.menu.main, menu);
            return true; } }

```

Pedoman.java

```

package com.ujicoba.pendaftarannamiklb;
import android.os.Bundle;
import android.app.Activity;
import android.content.Intent;
import android.view.Menu;
import android.view.View;

public class Pedoman extends Activity {
    @Override
        protected void onCreate(Bundle savedInstanceState) {
            super.onCreate(savedInstanceState);
            setContentView(R.layout.pedoman); }
    public void move(View view){
    Intent intent = new Intent(Pedoman.this, Pasphoto.class);
        startActivity(intent); }
    @Override
        public boolean onCreateOptionsMenu(Menu menu) {
            getMenuInflater().inflate(R.menu.pedoman, menu);
            return true; } }

```

Pasphoto.java

```

package com.ujicoba.pendaftarannamiklb;

import java.io.BufferedReader;

```

```

import java.io.ByteArrayInputStream;
import java.io.ByteArrayOutputStream;
import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.util.ArrayList;
import java.util.List;
import java.util.Random;
import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;
import org.apache.http.client.ClientProtocolException;
import org.apache.http.client.HttpClient;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.entity.mime.content.InputStreamBody;
import org.apache.http.entity.mime.content.StringBody;
import org.apache.http.impl.client.DefaultHttpClient;
import org.apache.http.message.BasicNameValuePair;
import org.apache.http.params.CoreProtocolPNames;
import org.apache.http.util.EntityUtils;
import org.json.JSONException;
import org.json.JSONObject;
import android.R.integer;
import android.annotation.SuppressLint;
import android.app.ActionBar;
import android.app.Activity;
import android.app.ProgressDialog;
import android.content.Intent;
import android.database.Cursor;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Bitmap.CompressFormat;
import android.net.Uri;
import android.os.AsyncTask;
import android.os.Bundle;
import android.os.StrictMode;
import android.provider.MediaStore;
import android.util.Log;
import android.view.MenuItem;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

public class Pasphoto extends Activity implements OnClickListener{
    String imagePath;
    String imageName;
    ImageView imgeView;
    long imageSize = 0; // kb
    // Progress Dialog

```

```

private ProgressDialog pDialog;
final static int REQUEST_CODE = 1;
JSONParser jsonParser = new JSONParser();
EditText inputName;
TextView status;
Button upload, next ;
URLConnection connection = null;
DataOutputStream outputStream = null;
DataInputStream inputStream = null;
Bitmap bm;
String user = null;
static String pathToOurFile = "", format;
public String Koneksi, isi;
String urlServer = "";
String lineEnd = "\r\n", twoHyphens = "--", boundary = "*****";
int nilai = (int) (Math.random() * 9999999);
int bytesRead, bytesAvailable, bufferSize;
byte[] buffer;
int maxBufferSize = 1 * 1024 * 1024;
private static String url_tambah_pendaftaran = "";
private static final String TAG_SUCCESS = "success";

@SuppressLint("NewApi")
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.foto);
    url_tambah_pendaftaran = "http://192.168.43.226/pendaftaran/program/pasfoto/simpan.php";
    urlServer = "http://192.168.43.226/pendaftaran/program/pasfoto/upload.php";
    StrictMode.ThreadPolicy policy = new
    StrictMode.ThreadPolicy.Builder()
        .permitAll().build();
    StrictMode.setThreadPolicy(policy);
    inputName = (EditText) findViewById(R.id.editnama);
    imageView = (ImageView) findViewById(R.id.imageView1);
    Button btnsimpan = (Button) findViewById(R.id.simpan);
    next = (Button) findViewById(R.id.lanjut);
    next.setVisibility(View.GONE);
    upload = (Button) findViewById(R.id.pilihimage);
    upload.setOnClickListener(this);
    btnsimpan.setOnClickListener(this);
    status = (TextView) findViewById(R.id.detail); }
@Override
public void onClick(View v) {
    switch (v.getId()) {
    case R.id.pilihimage:
        Intent intent = new Intent();
        intent.setType("image/*");
        intent.setAction(Intent.ACTION_GET_CONTENT);
        startActivityForResult(
            Intent.createChooser(intent, "Select Picture"),
            REQUEST_CODE);
        break;
    case R.id.simpan:
        if (this.imagePath == null) {
            // IF NO IMAGE SELECTED DO NOTHING
            Toast.makeText(this, "No image selected", Toast.LENGTH_SHORT)

```

```

        .show();
        return;}
        this.pDialog = this.createDialog();
        this.pDialog.show();
new ImageUploader().execute();
        break;
        default:
        break; } }
        @Override
protected void onActivityResult(int requestCode, int resultCode,
Intent data) {
super.onActivityResult(requestCode, resultCode, data);
        if (requestCode == REQUEST_CODE && resultCode == RESULT_OK) {
            Uri selectedImageUri = data.getData();
            imagePath = getPath(selectedImageUri);
            imageName = imagePath.substring(imagePath.lastIndexOf("."));
            imageSize = this.getFileSize(imagePath);
            Bitmap bitmap = BitmapFactory.decodeFile(imagePath);
            imageView.setImageBitmap(bitmap);
            status.setText("File path : " + imageName); }}
private String getPath(Uri uri) {
String[] projection = { MediaStore.Images.Media.DATA };
Cursor cursor = managedQuery(uri, projection, null, null, null);
int column_index = cursor
        .getColumnIndexOrThrow(MediaStore.Images.Media.DATA);
cursor.moveToFirst();
return cursor.getString(column_index);}

private ProgressDialog createDialog() {
ProgressDialog progressDialog = new ProgressDialog(this);
progressDialog.setMessage("Please wait.. Uploading File");
progressDialog.setProgressStyle(ProgressDialog.STYLE_HORIZONTAL);
progressDialog.setCancelable(false);
return progressDialog; }
private long getFileSize(String imagePath) {
long length = 0;
try { File file = new File(imagePath);
length = file.length();
length = length / 2048;
} catch (Exception e) {
e.printStackTrace(); }
return length; }
private class ImageUploader extends AsyncTask<Void, Integer,
Boolean>
implements UploadProgressListene {
@Override
protected Boolean doInBackground(Void... params) {
try { InputStream inputStream = new FileInputStream(new File(
imagePath));
byte[] data = this.convertToByteArray(inputStream);
HttpClient httpClient = new DefaultHttpClient();
httpClient.getParams().setParameter(
CoreProtocolNames.USER_AGENT, System.getProperty("http.agent"));
HttpPost httpPost = new HttpPost(urlServer);
StringBody dataString = new StringBody(
"This is the sample image");

InputStreamBody inputStreamBody = new InputStreamBody(
new ByteArrayInputStream(data), imageName);

```

```

multipartEntity = new MultipartEntity();
CustomMultiPartEntity multipartEntity = new
CustomMultiPartEntity();
    multipartEntity.setUploadProgressListener(this);
multipartEntity.addPart("file", inputStreamBody);
multipartEntity.addPart("description", dataString);
httpPost.setEntity(multipartEntity);
httpPost.setEntity(multipartEntity);
HttpResponse httpResponse = httpClient.execute(httpPost);
String stringResponse = EntityUtils.toString(httpResponse
.getEntity());
Log.d("data from server", stringResponse);
String name = inputName.getText().toString();
List<NameValuePair> param = new ArrayList<NameValuePair>();
param.add(new BasicNameValuePair("id_pendaftaran", name));
param.add(new BasicNameValuePair("foto", imageName));
JSONObject json = jsonParser.makeHttpRequest(
    url_tambah_pendaftaran, "POST", param);
Log.d("Create Response", json.toString());
} catch (FileNotFoundException e1) {
    e1.printStackTrace();
    return false;
} catch (ClientProtocolException e) {
    e.printStackTrace();
    return false;
} catch (IOException e) {
    e.printStackTrace();
    return false; }
return true; }
@Override
public void transferred(long num) {
    long dataUploaded = ((num / 2048) * 100) / imageSize;
    this.publishProgress((int) dataUploaded); }
private byte[] convertToByteArray(InputStream inputStream)
throws IOException {
    ByteArrayOutputStream bos = new ByteArrayOutputStream();
    int next = inputStream.read();
    while (next > -1) {
        bos.write(next);
        next = inputStream.read(); }
    bos.flush();
    return bos.toByteArray(); }
@Override
protected void onProgressUpdate(Integer... values) {
    super.onProgressUpdate(values);
    progressDialog.setProgress(values[0]); }
@Override
protected void onPostExecute(Boolean uploaded) {
    super.onPostExecute(uploaded);
    if (uploaded) {
        progressDialog.dismiss();
        Toast.makeText(Pasphoto.this, "File Uploaded",
            Toast.LENGTH_SHORT).show();
        next.setVisibility(View.VISIBLE);
    } else {
        progressDialog.setMessage("Uploading Failed");
        progressDialog.setCancelable(true);}}
public void go(View view){

```

```
Intent intent = new Intent(Pasphoto.this, Pendaftaran.class);
    startActivity(intent); } }
```

Pendaftaran.java

```
package com.ujicoba.pendaftarannamiklb;
import java.util.ArrayList;
import java.util.List;
import org.apache.http.NameValuePair;
import org.apache.http.message.BasicNameValuePair;
import org.json.JSONObject;
import android.os.AsyncTask;
import android.os.Bundle;
import android.app.Activity;
import android.app.ProgressDialog;
import android.content.Intent;
import android.view.Menu;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

public class Pendaftaran extends Activity {
    EditText id_pendaftaran,nama,jenis_kelamin,ttl,alamat,agama,
    hp,asal,tahun;
    Button submit,next;
    JSONParser jParser = new JSONParser();
    ProgressDialog pDialog;
    private static String url =
"http://192.168.43.226/pendaftaran/program/tambah.php";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.daftar);
        id_pendaftaran= (EditText)findViewById(R.id.id);
        nama= (EditText)findViewById(R.id.nama);
        jenis_kelamin= (EditText)findViewById(R.id.jenis);
        ttl=(EditText)findViewById(R.id.ttl);
        alamat = (EditText)findViewById(R.id.alamat);
        agama = (EditText)findViewById(R.id.agama);
        hp = (EditText)findViewById(R.id.hp);
        asal = (EditText)findViewById(R.id.asal);
        tahun = (EditText)findViewById(R.id.tahun);
        next = (Button) findViewById(R.id.lanjut);
        next.setVisibility(View.GONE);
        submit = (Button)findViewById(R.id.btnsubmit);
        submit.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View arg0) {
                new input().execute();} }); }
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if
it is present.
        getMenuInflater().inflate(R.menu.main, menu);
        return true; }
    public class input extends AsyncTask<String, String, String> {
        String success;
        @Override
        protected void onPreExecute() {
```

```

        super.onPreExecute();
        pDialog = new ProgressDialog(Pendaftaran.this);
        pDialog.setMessage("Lagi Proses Mohon Bersabar...");
        pDialog.setIndeterminate(false);
        pDialog.show(); }
@Override
protected String doInBackground(String... arg0) {
    String strId = id_pendaftaran.getText().toString();
    String strNama = nama.getText().toString();
    String strJenis = jenis_kelamin.getText().toString();
    String strTtl = ttl.getText().toString();
    String strAlamat = alamat.getText().toString();
    String strAgama = agama.getText().toString();
    String strHp = hp.getText().toString();
    String strAsal = asal.getText().toString();
    String strTahun = tahun.getText().toString();
    List<NameValuePair> params = new ArrayList<NameValuePair>();
    params.add(new BasicNameValuePair("id_pendaftaran", strId));
    params.add(new BasicNameValuePair("nama", strNama));
    params.add(new BasicNameValuePair("jenis_kelamin", strJenis));
    params.add(new BasicNameValuePair("ttl", strTtl));
    params.add(new BasicNameValuePair("alamat", strAlamat));
    params.add(new BasicNameValuePair("agama", strAgama));
    params.add(new BasicNameValuePair("hp", strHp));
    params.add(new BasicNameValuePair("asal", strAsal));
    params.add(new BasicNameValuePair("tahun", strTahun));
    JSONObject json = jParser.makeHttpRequest(url, "POST", params);
    try { success = json.getString("success");
        } catch (Exception e) {
            Toast.makeText(getApplicationContext(), "Error",
                Toast.LENGTH_LONG).show(); }
    return null; }
protected void onPostExecute(String file_url) {
    // dismiss the dialog once done
    pDialog.dismiss();
    if (success.equals("1")) {
        Toast.makeText(getApplicationContext(), "Biodata
Diri Anda Tersimpan!!!", Toast.LENGTH_LONG).show();
        next.setVisibility(View.VISIBLE);
        id_pendaftaran.setText("");
        nama.setText("");
        jenis_kelamin.setText("");
        ttl.setText("");
        alamat.setText("");
        agama.setText("");
        hp.setText("");
        asal.setText("");
        tahun.setText("");
    } else {
        Toast.makeText(getApplicationContext(), "Gagal Mohon
Diperiksa Kembali Data Anda!!!", Toast.LENGTH_LONG).show(); }}}
public void go(View view){
Intent intent = new Intent(Pendaftaran.this, Ijazah.class);
startActivity(intent); }}

```

Ijazah.java

```

package com.ujicoba.pendaftarannamiklb;
import java.io.BufferedReader;
import java.io.ByteArrayInputStream;

```



```

import java.io.ByteArrayOutputStream;
import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.util.ArrayList;
import java.util.List;
import java.util.Random;
import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;
import org.apache.http.client.ClientProtocolException;
import org.apache.http.client.HttpClient;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.entity.mime.content.InputStreamBody;
import org.apache.http.entity.mime.content.StringBody;
import org.apache.http.impl.client.DefaultHttpClient;
import org.apache.http.message.BasicNameValuePair;
import org.apache.http.params.CoreProtocolPNames;
import org.apache.http.util.EntityUtils;
import org.json.JSONException;
import org.json.JSONObject;
import android.R.integer;
import android.annotation.SuppressLint;
import android.app.ActionBar;
import android.app.Activity;
import android.app.ProgressDialog;
import android.content.Intent;
import android.database.Cursor;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.graphics.Bitmap.CompressFormat;
import android.net.Uri;
import android.os.AsyncTask;
import android.os.Bundle;
import android.os.StrictMode;
import android.provider.MediaStore;
import android.util.Log;
import android.view.MenuItem;
import android.view.View;
import android.view.View.OnClickListener;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

public class Ijazah extends Activity implements OnClickListener {
    String imagePath;
    String imageName;
    ImageView imggeView;
    long imageSize = 0; // kb
    // Progress Dialog
    private ProgressDialog pDialog;

```

```

final static int REQUEST_CODE = 1;
JSONParser jsonParser = new JSONParser();
EditText inputName;
TextView status;
Button upload, next ;
URLConnection connection = null;
DataOutputStream outputStream = null;
DataInputStream inputStream = null;
Bitmap bm;
String user = null;
static String pathToFile = "", format;
public String koneksi, isi;
String urlServer = "";
String lineEnd = "\r\n", twoHyphens = "--", boundary = "*****";
int nilai = (int) (Math.random() * 9999999);
int bytesRead, bytesAvailable, bufferSize;
byte[] buffer;
int maxBufferSize = 1 * 1024 * 1024;
private static String url_tambah_pendaftaran = "";
private static final String TAG_SUCCESS = "success";

@SuppressWarnings("NewApi")
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_ijazah);
    url_tambah_pendaftaran = "http://192.168.43.226/pendaftaran/program/ijazah/simpan.php";
    urlServer = "http://192.168.43.226/pendaftaran/program/ijazah/upload.php";
    StrictMode.ThreadPolicy policy = new StrictMode.ThreadPolicy.Builder().
        .permitAll().build();
    StrictMode.setThreadPolicy(policy);
    inputName = (EditText) findViewById(R.id.editnama);
    imageView = (ImageView) findViewById(R.id.imageView1);
    Button btnsimpan = (Button) findViewById(R.id.simpan);
    next = (Button) findViewById(R.id.lanjut);
    next.setVisibility(View.GONE);
    upload = (Button) findViewById(R.id.pilihimage);
    upload.setOnClickListener(this);
    btnsimpan.setOnClickListener(this);
    status = (TextView) findViewById(R.id.detail); }
@Override
public void onClick(View v) {
    switch (v.getId()) {
        case R.id.pilihimage:
            Intent intent = new Intent();
            intent.setType("image/*");
            intent.setAction(Intent.ACTION_GET_CONTENT);
            startActivityForResult(
                Intent.createChooser(intent, "Select Picture"),
                REQUEST_CODE);
            break;
        case R.id.simpan:
            if (this.imagePath == null) {
                // IF NO IMAGE SELECTED DO NOTHING
                Toast.makeText(this, "No image selected", Toast.LENGTH_SHORT)
                    .show();
            }
    }
}

```

```

        return;}
        this.pDialog = this.createDialog();
        this.pDialog.show();
new ImageUploader().execute();
        break;
        default:
        break; } }
@Override
protected void onActivityResult(int requestCode, int resultCode,
Intent data) {
super.onActivityResult(requestCode, resultCode, data);
if (requestCode == REQUEST_CODE && resultCode == RESULT_OK) {
Uri selectedImageUri = data.getData();
imagePath = getPath(selectedImageUri);
imageName = imagePath.substring(imagePath.lastIndexOf("."));
imageSize = this.getFileSize(imagePath);
Bitmap bitmap = BitmapFactory.decodeFile(imagePath);
imageView.setImageBitmap(bitmap);
status.setText("File path : " + imageName); }}
private String getPath(Uri uri) {
String[] projection = { MediaStore.Images.Media.DATA };
Cursor cursor = managedQuery(uri, projection, null, null, null);
int column_index = cursor
.getColumnIndexOrThrow(MediaStore.Images.Media.DATA);
cursor.moveToFirst();
return cursor.getString(column_index);}

private ProgressDialog createDialog() {
ProgressDialog progressDialog = new ProgressDialog(this);
progressDialog.setMessage("Please wait.. Uploading File");
progressDialog.setProgressStyle(ProgressDialog.STYLE_HORIZONTAL);
progressDialog.setCancelable(false);
return progressDialog; }
private long getFileSize(String imagePath) {
long length = 0;
try { File file = new File(imagePath);
length = file.length();
length = length / 2048;
} catch (Exception e) {
e.printStackTrace(); }
return length; }
private class ImageUploader extends AsyncTask<Void, Integer,
Boolean>
implements UploadProgressListene {
@Override
protected Boolean doInBackground(Void... params) {
try { InputStream inputStream = new FileInputStream(new File(
imagePath));
byte[] data = this.convertToByteArray(inputStream);
HttpClient httpClient = new DefaultHttpClient();
httpClient.getParams().setParameter(
CoreProtocolNames.USER_AGENT, System.getProperty("http.agent"));
HttpPost httpPost = new HttpPost(urlServer);
StringBody dataString = new StringBody(
"This is the sample image");

InputStreamBody inputStreamBody = new InputStreamBody(
new ByteArrayInputStream(data), imageName);

```

```

multipartEntity = new MultipartEntity();
CustomMultiPartEntity multipartEntity = new
    CustomMultiPartEntity();
    multipartEntity.setUploadProgressListener(this);
    multipartEntity.addPart("file", inputStreamBody);
    multipartEntity.addPart("description", dataString);
    httpPost.setEntity(multipartEntity);
    httpPost.setEntity(multipartEntity);
    HttpResponse httpResponse = httpClient.execute(httpPost);
    String stringResponse = EntityUtils.toString(httpResponse
        .getEntity());
    Log.d("data from server", stringResponse);
    String name = inputName.getText().toString();
    List<NameValuePair> param = new ArrayList<NameValuePair>();
    param.add(new BasicNameValuePair("id_pendaftaran", name));
    param.add(new BasicNameValuePair("file_ijazah", imageName));
    JSONObject json = jsonParser.makeHttpRequest(
        url_tambah_pendaftaran, "POST", param);
    Log.d("Create Response", json.toString());
} catch (FileNotFoundException e1) {
    e1.printStackTrace();
    return false;
} catch (ClientProtocolException e) {
    e.printStackTrace();
    return false;
} catch (IOException e) {
    e.printStackTrace();
    return false; }
return true; }
@Override
public void transferred(long num) {
    long dataUploaded = ((num / 2048) * 100) / imageSize;
    this.publishProgress((int) dataUploaded); }
private byte[] convertToByteArray(InputStream inputStream)
    throws IOException {
    ByteArrayOutputStream bos = new ByteArrayOutputStream();
    int next = inputStream.read();
    while (next > -1) {
        bos.write(next);
        next = inputStream.read(); }
    bos.flush();
    return bos.toByteArray(); }
@Override
protected void onProgressUpdate(Integer... values) {
    super.onProgressUpdate(values);
    progressDialog.setProgress(values[0]); }
@Override
protected void onPostExecute(Boolean uploaded) {
    super.onPostExecute(uploaded);
    if (uploaded) {
        progressDialog.dismiss();
        Toast.makeText(Ijazah.this, "Ijazah Terupload",
            Toast.LENGTH_SHORT).show();
        next.setVisibility(View.VISIBLE);
    } else {
        progressDialog.setMessage("Uploading Failed");
        progressDialog.setCancelable(true); }}}
public void go(View view){
Intent intent = new Intent(Ijazah.this, Selesai.class);

```

```
startActivity(intent); }}
```

Selesai.java

```
package com.ujicoba.pendaftarannamiklb;
import android.os.Bundle;
import android.app.Activity;
import android.content.Intent;
import android.view.Menu;
import android.view.View;

public class Selesai extends Activity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.selesai); }
    public void done(View view){
Intent intent = new Intent(Selesai.this, MainActivity.class);
    startActivity(intent); }
    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.selesai, menu);
        return true; }}
```

activity_splashscreen.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="#ff8c00" >
    <RelativeLayout
        android:id="@+id/rl"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentRight="true"
        android:layout_alignParentTop="true" >
        <TextView
            android:id="@+id/textV"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_below="@+id/splashimg"
            android:layout_centerHorizontal="true"
            android:gravity="center_horizontal|top"
            android:height="40dp"
            android:paddingLeft="10dp"
            android:paddingRight="10dp"
            android:text="@string/aplikasi"
            android:textAppearance="?android:attr/textAppearanceLarge"
            android:textColor="#fff"
            android:textSize="16sp"
            android:textStyle="bold"
            android:typeface="normal" />
        <ProgressBar
            android:id="@+id/progressBar1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@+id/textV"
```

```

        android:layout_centerHorizontal="true" />
<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/progressBar1"
    android:layout_centerHorizontal="true"
    android:text="@string/wait"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:textColor="#fff" />
<ImageView
    android:id="@+id/splashing"
    android:layout_width="match_parent"
    android:layout_height="200dp"
    android:layout_alignParentLeft="true"
    android:layout_alignParentTop="true"
    android:layout_marginTop="38dp"
    android:src="@drawable/amik_lb2000" />
</RelativeLayout>
</RelativeLayout>

```

activity_main.xml

```

<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MainActivity"
    android:background="#ff8c00" >
<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:text="@string/welcome"
    android:textAppearance="?android:attr/textAppearanceLarge" />
<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/textView1"
    android:layout_centerHorizontal="true"
    android:text="@string/yulb"
    android:textAppearance="?android:attr/textAppearanceLarge" />
<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/textView2"
    android:layout_centerHorizontal="true"
    android:text="@string/ulb"
    android:textAppearance="?android:attr/textAppearanceLarge" />
<Button
    android:id="@+id/buttonBiru"

```

```

        style="@style/ButtonText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignBottom="@+id/imageView1"
        android:layout_centerHorizontal="true"
        android:background="@drawable/buttonbiru"
        android:onClick="move"
        android:text="@string/daftar" />
<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_above="@+id/buttonBiru"
    android:layout_alignLeft="@+id/buttonBiru"
    android:text="@string/kuy"
android:textAppearance="?android:attr/textAppearanceMedium" />
<ImageView
    android:id="@+id/imageView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView2"
    android:layout_alignRight="@+id/textView3"
    android:layout_alignTop="@+id/textView1"
    android:src="@drawable/amik_lb2000" />
</RelativeLayout>

```

pedoman.xml

```

<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".Pedoman" >
<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:text="@string/petunjuk"
android:textAppearance="?android:attr/textAppearanceLarge" />
<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/textView1"
    android:layout_marginTop="31dp"
    android:text="@string/rule1" />
<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignRight="@+id/textView2"
    android:layout_below="@+id/textView2"

```

```

        android:layout_marginTop="24dp"
        android:text="@string/rule2" />
<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_centerVertical="true"
    android:text="@string/contoh"
android:textAppearance="?android:attr/textAppearanceMedium" />
<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView1"
    android:layout_below="@+id/textView4"
    android:text="@string/arie"
android:textAppearance="?android:attr/textAppearanceMedium" />
<TextView
    android:id="@+id/textView6"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/textView5"
    android:layout_marginTop="15dp"
    android:text="@string/jadi"
android:textAppearance="?android:attr/textAppearanceMedium" />
<TextView
    android:id="@+id/textView7"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/textView6"
    android:layout_centerHorizontal="true"
    android:text="@string/ap"
android:textAppearance="?android:attr/textAppearanceMedium" />
<TextView
    android:id="@+id/textView8"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_centerHorizontal="true"
    android:layout_marginBottom="56dp"
    android:text="@string/mulai"
android:textAppearance="?android:attr/textAppearanceMedium" />
<Button
    android:id="@+id/button1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_alignRight="@+id/textView3"
    android:layout_alignTop="@+id/textView8"
    android:layout_marginTop="17dp"
    android:onClick="move"
    android:text="@string/start" />
</RelativeLayout>

```

foto.xml


```

<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >
<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/pasfto"
android:textAppearance="?android:attr/textAppearanceLarge" />
<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/psft"
android:textAppearance="?android:attr/textAppearanceMedium" />
<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/rule" />
<EditText
    android:id="@+id/editnama"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:hint="Id Pendaftaran"
    android:ems="10" >
    <requestFocus />
</EditText>
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content" >
<Button
    android:id="@+id/pilihimage"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="@string/pilih" />
<TextView
    android:id="@+id/detail"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="TextView" />
</LinearLayout>
<ImageView
    android:id="@+id/imageView1"
    android:layout_width="208dp"
    android:layout_height="184dp"
    android:scaleType="centerCrop"
    android:src="@drawable/ic_launcher" />
<Button
    android:id="@+id/simpan"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Upload" />
<Button
    android:id="@+id/lanjut"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"

```

```

        android:onClick="go"
        android:visibility="gone"
        android:text="NEXT >>" />
</LinearLayout>

```

daftar.xml

```

<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical" >
    <ScrollView
        android:layout_height="match_parent"
        android:layout_width="match_parent"
        android:scrollbarStyle="insideInset">
        <LinearLayout
            android:layout_height="match_parent"
            android:layout_width="match_parent"
            android:orientation="vertical">
            <TextView
                android:id="@+id/textView1"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:text="@string/bio"
                android:textAppearance="?android:attr/textAppearanceLarge" />
            <EditText
                android:id="@+id/id"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginLeft="10dp"
                android:layout_marginTop="10dp"
                android:hint="ID Pendaftaran"
                android:paddingRight="10dp" />
            <EditText
                android:id="@+id/nama"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginLeft="10dp"
                android:layout_marginTop="10dp"
                android:hint="Nama Lengkap"
                android:paddingRight="10dp" />
            <EditText
                android:layout_height="wrap_content"
                android:layout_width="match_parent"
                android:id="@+id/jenis"
                android:layout_marginTop="10dp"
                android:layout_marginLeft="10dp"
                android:paddingRight="10dp"
                android:hint="Jenis Kelamin (Laki-laki/Perempuan)" />
            <EditText
                android:id="@+id/ttl"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginLeft="10dp"
                android:layout_marginTop="10dp"
                android:hint="Tempat/Tanggal Lahir"
                android:paddingRight="10dp" />
            <EditText
                android:layout_height="wrap_content"

```

```

        android:layout_width="match_parent"
        android:id="@+id/agama"
        android:layout_marginTop="10dp"
        android:layout_marginLeft="10dp"
        android:paddingRight="10dp"
        android:hint="Agama"/>

<EditText
    android:layout_height="wrap_content"
    android:layout_width="match_parent"
    android:id="@+id/alamat"
    android:layout_marginTop="10dp"
    android:layout_marginLeft="10dp"
    android:paddingRight="10dp"
    android:hint="Alamat"/>
<EditText
    android:id="@+id/hp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="10dp"
    android:layout_marginTop="10dp"
    android:hint="No Handphone"
    android:paddingRight="10dp" />
<EditText
    android:id="@+id/asal"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="10dp"
    android:layout_marginTop="10dp"
    android:hint="Asal Sekolah"
    android:paddingRight="10dp" />
<EditText
    android:id="@+id/tahun"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="10dp"
    android:layout_marginTop="10dp"
    android:hint="Tahun Tamat"
    android:paddingRight="10dp" />
<Button
    android:id="@+id/btnsubmit"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginLeft="10dp"
    android:layout_marginTop="10dp"
    android:paddingRight="10dp"
    android:text="@string/save" />
<Button
    android:id="@+id/lanjut"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:onClick="go"
    android:visibility="gone"
    android:text="NEXT >>" />
</LinearLayout>
</ScrollView>
</LinearLayout>

```

activity_ijazah.xml

```

<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical" >
    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/up"
    android:textAppearance="?android:attr/textAppearanceLarge" />
    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/scan"
    android:textAppearance="?android:attr/textAppearanceMedium" />
    <TextView
        android:id="@+id/textView2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/note" />
    <EditText
        android:id="@+id/editnama"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Id Pendaftaran"
        android:ems="10" >
        <requestFocus />
    </EditText>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content" >
        <Button
            android:id="@+id/pilihimage"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="@string/choose" />
        <TextView
            android:id="@+id/detail"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="TextView" />
    </LinearLayout>
    <ImageView
        android:id="@+id/imageView1"
        android:layout_width="208dp"
        android:layout_height="184dp"
        android:scaleType="centerCrop"
        android:src="@drawable/ic_launcher" />
    <Button
        android:id="@+id/simpan"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Upload" />
    <Button
        android:id="@+id/lanjut"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"

```

```

        android:onClick="go"
        android:visibility="gone"
        android:text="PENDAFTARAN SELESAI" />
</LinearLayout>

```

selesai.xml

```

<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:paddingBottom="@dimen/activity_vertical_margin"
android:paddingLeft="@dimen/activity_horizontal_margin"
android:paddingRight="@dimen/activity_horizontal_margin"
android:paddingTop="@dimen/activity_vertical_margin"
tools:context=".MainActivity"
android:background="#ff8c00" >
<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/textView1"
    android:layout_centerHorizontal="true"
    android:text="@string/telah"
    android:textAppearance="?android:attr/textAppearanceLarge" />
<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/textView2"
    android:layout_centerHorizontal="true"
    android:text="@string/ulb"
    android:textAppearance="?android:attr/textAppearanceLarge" />
<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_centerHorizontal="true"
    android:text="@string/thanks"
    android:textAppearance="?android:attr/textAppearanceLarge" />
<Button
    android:id="@+id/button1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentRight="true"
    android:onClick="done"
    android:text="@string/finish" />
<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_above="@+id/button1"
    android:layout_alignLeft="@+id/button1"
    android:text="@string/kelanjutan"
    android:textAppearance="?android:attr/textAppearanceLarge" />
<ImageView

```

```

        android:id="@+id/imageView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignLeft="@+id/button1"
        android:layout_alignParentTop="true"
        android:src="@drawable/amik_lb2000" />
</RelativeLayout>

```

Web - > Admin

koneksi.php

```

<?php
$host = "localhost";
$user = "root";
$pass = "";
$name = "mahasiswabarur";
$koneksi = mysql_connect($host,$user,$pass) or
die("koneksi ke database gagal!");
mysql_select_db($name,$koneksi) or die
("Tidak ada database yang dipilih!"); ?>

```

conn.php

```

<?php
session_start();
$koneksi=new mysqli("localhost","root","","mahasiswabarur"); ?>

```

index.php

```

<!DOCTYPE html>
<html lang="en">
<head>
    <title>ADMIN AMIK LABUHANBATU</title>
    <meta charset="UTF-8">
</head>
<body>
<form method="post" class="login100-form validate-form"
action="login.php">
<span class="login100-form-title p-b-34">
    ADMIN AMIK LABUHANBATU LOGIN
    <br>Untuk Mengakses Data<br>
    Masukkan Username dan password
</span>
<div class="wrap-input100 rs1-wrap-input100 validate-input m-b-20"
data-validate="Type user name">
<input id="first-name" class="input100" type="text"
name="username" placeholder="User name">
<span class="focus-input100"></span> </div>
<div class="wrap-input100 rs2-wrap-input100 validate-input m-b-20"
data-validate="Type password">
<input class="input100" type="password" name="pass"
placeholder="Password">
<span class="focus-input100"></span></div>
<div class="container-login100-form-btn">
<input name="login" type="submit" value="LOGIN" class="btn btn-
primary btn-block"> </div> </form>
<div class="login100-more" style="background-image:
url('images/cover_login.png');"></div>
</div></div> </div>
</body></html>

```

login.php

```

<?php
extract($_POST);
include 'conn.php';
if (isset($_POST['login'])) {
$ambil = $koneksi->query("SELECT * FROM admin WHERE
username='$_POST[username]' AND password='$_POST[pass]'");
$yangcocok = $ambil->num_rows;
if ($yangcocok==1) {
$_SESSION['admin']=$ambil->fetch_assoc();
echo "<script>alert('LOGIN SUKSES!!! Selamat Datang Admin AMIK
Labuhanbatu');</script>";
echo "<meta http-equiv='refresh' content='1;url=home.php'>";
} else {
echo "<script>alert('Login gagal!! Periksa Kembali Username &
Password!');</script>";
echo "<meta http-equiv='refresh' content='1;url=index.php'>";}}?>

```

home.php

```

<head>
<title>ADMIN AMIK LABUHANBATU</title>
<!-- The fav icon -->
<link rel="shortcut icon" href="img/amik_lb.png"></head>
<body>
<a class="navbar-brand" href="index.html">
<small><span>ADMIN AMIK LABUHANBATU</span></small></a></div>
</div>
<ul class="nav nav-pills nav-stacked main-menu">
<li class="nav-header">MENU NAVIGASI</li>
<li><a class="ajax-link" href="home.php"><i class="glyphicon
glyphicon-home"></i><span> HOME</span></a></li>
<li><a class="ajax-link" href="calonmhs.php"><i class="glyphicon
glyphicon-edit"></i><span> Data Calon Mahasiswa </span></a></li>
<li><a href="logout.php"><i class="glyphicon glyphicon-ban-
circle"></i><span> LOG OUT</span></a></li></ul>
</div>
<div class="row">
<div class="box col-md-12">
<div class="box-inner">
<div class="box-header well">
<h2><i class="glyphicon glyphicon-info-sign"></i> HOME</h2>
<div class="box-icon">
<a href="#" class="btn btn-minimize btn-round btn-default"><i
class="glyphicon glyphicon-chevron-up"></i></a></div></div>
<div class="box-content row">
<div class="col-lg-7 col-md-12">
<h1>SELAMAT DATANG <br>

<small>Admin AMIK LABUHANBATU.</small></h1>
<p><b>Silahkan Untuk Mengelola Data Tentang Calon Mahasiswa Baru
yang sudah mendaftar di AMIK LABUHANBATU</b></p> </div>
<!-- Ads, you can remove these -->
</div></div></div></div>
<!-- Ad ends --><hr></div>
<footer class="row">
<p class="col-md-9 col-sm-9 col-xs-12 copyright">&copy; <a
href="http://amik-labuhanbatu.ac.id" target="_blank">AMIK
LABUHANBATU</a> 2019</p>

```

```

        <p class="col-md-3 col-sm-3 col-xs-12 powered-by">Didesain
Oleh: <a href="/arie_pramana/">ARIE PRAMANA</a></p>
    </footer>
</div><!--/.fluid-container-->

```

calonmhs.php

```

<head>
    <title>ADMIN AMIK LABUHANBATU</title>
    <!-- The fav icon -->
    <link rel="shortcut icon" href="img/amik_lb.png"></head>
<body>
    <a class="navbar-brand" href="index.html">
<small><span>ADMIN AMIK LABUHANBATU</span></small></a></div>
</div>
    <ul class="nav nav-pills nav-stacked main-menu">
<li class="nav-header">MENU NAVIGASI</li>
<li><a class="ajax-link" href="home.php"><i class="glyphicon
glyphicon-home"></i><span> HOME</span></a></li>
<li><a class="ajax-link" href="calonmhs.php"><i class="glyphicon
glyphicon-edit"></i><span> Data Calon Mahasiswa </span></a></li>
<li><a href="logout.php"><i class="glyphicon glyphicon-ban-
circle"></i><span> LOG OUT</span></a></li></ul>
</div>
<div class="row">
    <div class="box col-md-12">
        <div class="box-inner">
            <div class="box-header well" data-original-title="">
                <h2><i class="glyphicon glyphicon-edit"></i> Data Calon
Mahasiswa</h2>
                <div class="box-icon">
                    <a href="#" class="btn btn-minimize btn-round btn-default"><i
class="glyphicon glyphicon-chevron-up"></i></a></div></div>
                    <div class="content">
                        <div class="container-fluid">
                            <div class="row">
                                <div class="box-content">
                                    <div class="control-group">
<body>
    <table border="3">
        <th>No</th>
        <th>ID Pendaftaran</th>
        <th>Pasfoto</th>
        <th>Nama Lengkap</th>
        <th>Jenis Kelamin</th>
        <th>Tempat/Tanggal Lahir</th>
        <th>Alamat</th>
        <th>Agama</th>
        <th>No.Handphone</th>
        <th>Asal Sekolah</th>
        <th>Tahun Tamat</th>
        <th>Lampiran (Ijazah)</th>
        <?php
            mysql_connect("localhost", "root", "");
            mysql_select_db("mahasiswabar");
            $query="SELECT
bio.id_pendaftaran,bio.nama,bio.jenis_kelamin,bio.ttl,bio.alamat,b
io.agama,bio.hp,bio.asal,bio.tahun,pasfoto.foto,ijazah.file_ijazah
FROM (biodata bio left JOIN pasfoto pasfoto on bio.id_pendaftaran

```



```

= pasfoto.id_pendaftaran) LEFT JOIN ijazah ijazah on
pasfoto.id_pendaftaran = ijazah.id_pendaftaran order by nama ASC";
$hasil=mysql_query($query);
$no_urut=0;
while($data = mysql_fetch_array($hasil)) {
    $no_urut++;
    echo "<tr>
        <td><b>\".$no_urut.\"</b></td>
        <td><b>\".$data['id_pendaftaran'].\"</b></td>
        <td><img
src='/pendaftaran/program/pasfoto/img/\".$data['foto'].\"'
height='150' width='90' ></td>
        <td><b>\".$data['nama'].\"</b></td>
        <td><b>\".$data['jenis_kelamin'].\"</b></td>
        <td><b>\".$data['ttl'].\"</b></td>
        <td><b>\".$data['alamat'].\"</b></td>
        <td><b>\".$data['agama'].\"</b></td>
        <td><b>\".$data['hp'].\"</b></td>
        <td><b>\".$data['asal'].\"</b></td>
        <td><b>\".$data['tahun'].\"</b></td>
        <td><b><u><i><img
src='/pendaftaran/program/ijazah/img/\".$data['file_ijazah'].\"'
height='150' width='90' ><br><a
href='/pendaftaran/program/ijazah/img/\".$data['file_ijazah'].\"'>Pe
rbesar Ijazah</a></br><i><u></b></td></tr> ";} ?>
</table></body> </div></div>
<hr></div>
<footer class="row">
    <p class="col-md-9 col-sm-9 col-xs-12 copyright">&copy; <a
href="http://amik-labuhanbatu.ac.id" target="_blank">AMIK
LABUHANBATU</a> 2019</p>

    <p class="col-md-3 col-sm-3 col-xs-12 powered-by">Didesain
Oleh: <a
href="/arie_pramana/">ARIE PRAMANA</a></p>
</footer>

</div><!--/.fluid-container-->

```

logout.php

```

<?php
session_destroy();
echo "<script>alert('Anda Telah Logout!!!');</script>";
echo "<script>location='index.php';</script>";
?>

```

PERYATAAN

Yang bertandatangan dibawah ini :

Nama : Arie Pramana
NPM : 16.051.00.006
Judul Tugas Akhir : Aplikasi Pendaftaran Mahasiswa Baru Berbasis
Android (Studi Kasus : AMIK Labuhanbatu)

Dengan ini penulis menyatakan bahwa Tugas Akhir ini disusun sebagai syarat untuk memperoleh gelar Ahli Madya pada Program Studi Manajemen Informatika Fakultas Sains dan Teknologi Universitas Labuhanbatu adalah hasil karya tulis penulis sendiri. Semua kutipan maupun rujukan dalam penulisan Tugas Akhir ini telah penulis cantumkan sumbernya dengan benar sesuai dengan ketentuan yang berlaku.

Jika di kemudian hari ternyata ditemukan seluruh atau sebagian skripsi ini bukan hasil karya penulis atau plagiat, penulis bersedia menerima sanksi pencabutan gelar akademik yang disandang dan sanksi-sanksi lainnya sesuai dengan peraturan perundang-undangan yang berlaku.

Rantauprapat, 31 Juli 2019
yang membuat pernyataan,

ARIE PRAMANA
16.051.00.006

