

# Perancangan Sistem Pengolahan Data Pendaftar Seleksi Penerimaan Mahasiswa Baru Jalur Masuk UMPB Berbasis Web

Yesica Febri Susanti<sup>1\*</sup>, Nelmiawati<sup>2\*</sup>

\*Teknik Informatika, Politeknik Negeri Batam

[yesi.ica.eci@gmail.com](mailto:yesi.ica.eci@gmail.com)<sup>1</sup>, [mia@polibatam.ac.id](mailto:mia@polibatam.ac.id)<sup>2</sup>

## Article Info

### Article history:

Received ...

Revised ...

Accepted ...

### Keyword:

Sistem pengolahan data, Metode *prototyping*, *black box testing*, Oracle.

## ABSTRACT

Indonesia dengan populasi yang besar, menghadapi peningkatan permintaan terhadap perguruan tinggi dengan semakin banyak individu yang mencari akses ke lembaga pendidikan tinggi. Jalur mandiri sering menjadi opsi bagi yang tidak lolos seleksi nasional. Terlihat dari tingginya tingkat persaingan dalam upaya diterima di Politeknik Negeri Batam (Polibatam) yang memiliki alternatif seleksi mandiri untuk penerimaan mahasiswa baru, yang dikenal sebagai jalur Ujian Masuk Politeknik Batam (UMPB). Dengan jumlah pendaftar yang tidak sedikit, Polibatam perlu melakukan manajemen data pendaftar yang cermat, yang mencakup pengelolaan data pendaftar hingga penentuan hasil seleksi penerimaan pendaftar. Proses seleksi mahasiswa baru memegang peran krusial dalam menentukan calon mahasiswa yang akan diterima. Namun, saat ini proses seleksi penerimaan mahasiswa baru di Politeknik Negeri Batam masih bersifat manual dengan menggunakan Microsoft Excel. Dampak dari proses manual ini rapat kelulusan memerlukan waktu yang cukup lama dan memungkinkan terjadinya resiko kesalahan manusia. Oleh karena itu dibuat sebuah sistem pengolahan data pendaftar seleksi penerimaan mahasiswa baru jalur masuk UMPB berbasis web yang bertujuan untuk mengotomatisasi proses seleksi dalam rapat kelulusan yang dapat memproses data dengan cepat dan akurat, menghilangkan keterlibatan manual yang memakan waktu, serta mengurangi risiko kesalahan manusia. Perancangan sistem web ini dibangun dengan menggunakan Metode *Prototyping*. Implementasi dilakukan menggunakan teknologi web seperti HTML, CSS, JavaScript, dan PHP dengan basis data Oracle. Pengujian sistem dilakukan menggunakan Metode *Black Box* untuk menguji fungsionalitas sistem.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

## I. PENDAHULUAN

Indonesia dengan populasi yang besar menghadapi peningkatan permintaan terhadap perguruan tinggi, dengan semakin banyak individu yang mencari akses ke lembaga pendidikan tinggi. Jalur mandiri sering menjadi opsi bagi yang tidak lolos seleksi nasional. Situasi ini membuat tingginya tingkat persaingan dalam upaya diterima di Politeknik Negeri Batam (Polibatam) yang merupakan satu-satunya perguruan tinggi vokasi negeri di Provinsi Kepulauan Riau.

Politeknik Negeri Batam memiliki seleksi mandiri untuk penerimaan mahasiswa baru yang dikenal sebagai jalur Ujian Masuk Politeknik Batam (UMPB). Banyak calon mahasiswa

yang memilih untuk mendaftar melalui jalur UMPB. Data statistik pendaftaran seleksi mandiri pada tahun 2023 menunjukkan bahwa terdapat 69 pendaftar untuk kelas industri, 3.331 pendaftar UMPB reguler malam, dan 2.590 pendaftar UMPB reguler Pagi [1].

Dengan jumlah pendaftar yang tidak sedikit, proses seleksi mahasiswa baru memegang peran krusial dalam menentukan calon mahasiswa terbaik yang akan diterima. Keberhasilan dalam menjalankan proses seleksi mahasiswa baru menjadi faktor utama dalam menjaga kualitas pendidikan.

Permasalahan yang terjadi di Politeknik Negeri Batam adalah cara seleksi penerimaan mahasiswa baru yang masih bersifat manual dengan menggunakan Microsoft Excel untuk memberikan tanda kelulusan bagi setiap pendaftar. Dampak

dari pendekatan manual ini adalah rapat kelulusan memerlukan waktu yang cukup lama dalam menyelesaikan proses seleksi mahasiswa baru untuk setiap program studi. Selain itu, apabila suatu program studi tidak dapat hadir dalam rapat kelulusan, program studi tersebut harus mengirimkan daftar nama calon mahasiswa yang lulus kepada pihak Akademik. Situasi ini memungkinkan terjadinya penulisan nama pendaftar yang sudah tercatat lulus di program studi lain.

Berdasarkan permasalahan yang telah dijelaskan, maka dirancang sebuah sistem dengan judul “Perancangan Sistem Pengolahan Data Pendaftar Seleksi Penerimaan Mahasiswa Baru Jalur Masuk UMPB Berbasis Web”. Sistem ini bertujuan untuk memberikan solusi dari sisi teknologi dengan menerapkan teknologi informasi yang dapat mengotomatisasi proses seleksi penerimaan mahasiswa baru di Politeknik Negeri Batam. Dengan merancang sebuah sistem pengolahan data pendaftar, aplikasi ini akan berintegrasi dengan Moodle pelaksanaan UMPB dan memanfaatkan algoritma yang telah diprogram dengan baik untuk menjalankan proses seleksi secara otomatis. Dengan mengotomatisasi proses seleksi, sistem ini dapat memproses data dengan cepat dan akurat, menghilangkan keterlibatan manual yang memakan waktu, serta mengurangi risiko kesalahan manusia. Sistem ini dibangun menggunakan bahasa pemrograman PHP dengan basis data Oracle.

## II. LANDASAN TEORI

### A. Tinjauan Pustaka

Dalam perancangan sistem pengolahan data pendaftar seleksi penerimaan mahasiswa baru ini terdapat beberapa penelitian sebelumnya yang telah dibuat dengan berbagai pengembangan yang berbeda-beda sesuai kebutuhannya, seperti yang dilakukan oleh Desty Dwi Pertiwi dan Rohmat Taufiq dengan judul “Analisis Dan Desain Sistem Informasi Pengolahan Nilai Siswa Di SMK Avicena Rajeg”, sistem ini dibuat untuk menunjang kinerja para wali kelas dan guru sehingga kemudahan, kecepatan, dan ketepatan dalam pengolahan nilai siswa dapat terlaksana [2].

Selanjutnya penelitian kedua dilakukan oleh Widiawati dan Hairil Kurniadi Siradjuddin yang berjudul “Perancangan Website Sistem Seleksi Siswa Baru menggunakan Framework CodeIgniter Pada Madrasah Aliyah Alkhairaat Kalumpang Kota Ternate”, sistem ini dibuat untuk mempermudah pihak sekolah dalam mengelola data pendaftaran, seleksi, dan menginformasikan pengumuman kelulusan [3].

Selanjutnya penelitian ketiga dilakukan oleh Yuda Irawan dan Refni Wahyuni yang berjudul “Sistem Pendukung Keputusan Penerimaan Siswa Baru di SMK Negeri 1 Tapung Hulu Menggunakan Metode Simple Multi Attribut Rating Technique (SMART)”, tujuan utama sistem ini adalah meningkatkan efisiensi pengolahan data dalam proses pengambilan keputusan untuk menentukan penerimaan atau penolakan calon siswa baru [4].

### B. Pengolahan Data

Pengolahan data merupakan proses yang dilakukan dalam hal mengumpulkan, mengatur dan menganalisis data atau informasi, yang bertujuan untuk mengubah data mentah menjadi informasi yang berguna dan lebih bermanfaat [5].

### C. UMPB

Ujian Masuk Politeknik Batam (UMPB) adalah sebuah proses seleksi yang dilakukan oleh Politeknik Negeri Batam secara independen. Dalam proses ini, pendaftar diharuskan mengikuti ujian tertulis yang telah diselenggarakan oleh institusi tersebut. Perlu diperhatikan bahwa pendaftar UMPB hanya dapat memilih program studi yang ditawarkan oleh Politeknik Negeri Batam dan sesuai dengan jenis program yang di inginkan. Program studi tersebut mencakup ujian masuk untuk program studi D3 dan D4 reguler pada kelas pagi, D3 dan D4 reguler pada kelas malam, serta program kelas industri [6].

### D. Web

*World Wide Web* (WWW), atau sering disebut web, merupakan salah satu aset internet yang sedang berkembang dengan pesat. Informasi di web didistribusikan melalui tautan (*hyperlinks*), yang memungkinkan pengguna untuk mengakses informasi dengan melompat dari satu halaman ke halaman lainnya [5].

### E. PHP

PHP, singkatan dari *Hypertext Preprocessor*, adalah salah satu bahasa pemrograman *open source* yang sangat sesuai untuk pengembangan web dan dapat disisipkan ke dalam dokumen HTML. PHP adalah bahasa skrip sisi server, di mana pengolahan data terjadi di server. Dengan kata lain, server akan mengeksekusi skrip program, dan hasilnya baru akan dikirimkan kepada klien yang melakukan permintaan. PHP yaitu bahasa pemrograman berbasis skrip yang digunakan untuk memproses data dan mengirimkannya kembali ke peramban web dalam bentuk kode HTML [7].

Sistem ini dibuat menggunakan bahasa pemrograman PHP. Kode PHP disisipkan langsung ke dalam dokumen HTML, kemudian diapit oleh tag, yang memungkinkan untuk beralih antara mode HTML dan PHP dengan mudah. PHP dapat dengan mudah mengambil data, memvalidasi input, dan menyimpan atau mengirim data ke basis data.

### F. Oracle

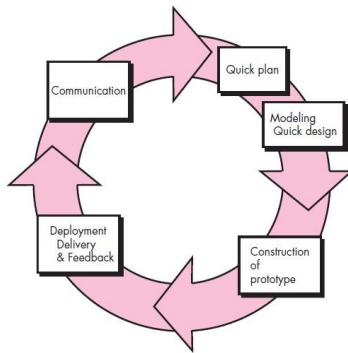
Oracle adalah *Relational Database Management System* (RDBMS) yang digunakan untuk mengelola informasi dengan cara yang terbuka, menyeluruh, dan terintegrasi. Oracle Server menyajikan solusi yang sangat efisien dan efektif dalam pemrosesan data [8]. Basis data Oracle memiliki keunggulan dibandingkan dengan DBMS lain dalam berbagai situasi. Fitur manajemen memori yang canggih dan opsi optimasi yang mendorong efisiensi sistem memungkinkan

perusahaan untuk memproses data dalam jumlah besar dengan cepat dan akurat [9].

Sistem ini dibangun menggunakan basis data Oracle. Basis data Oracle menggunakan SQL sebagai bahasa query untuk mengakses dan mengelola data. Dengan SQL, dapat melakukan operasi seperti *select*, *insert*, *update*, *delete*, dan lainnya.

### III. METODE

Metode pengembangan sistem pengolahan data hasil seleksi penerimaan mahasiswa baru jalur masuk UMPB ini dibangun dengan menggunakan Metode *Prototyping*. Metode *Prototyping* adalah suatu pendekatan dalam pengembangan perangkat lunak yang sering dimanfaatkan oleh pengembang untuk berinteraksi dengan pengguna selama tahap pembuatan sistem [10]. Tahapan Metode *Prototyping* digambarkan pada Gambar 1.



Gambar 1. Metode *Prototyping*

Metode ini memiliki beberapa tahapan, adapun tahapan yang dilakukan dalam Metode *Prototyping* yaitu:

#### A. Komunikasi (*Communication*)

Pada tahap ini dilakukan analisis kebutuhan dasar sistem yang akan dirancang dengan melakukan pengumpulan informasi kebutuhan sistem. Informasi dikumpulkan melalui diskusi dengan pihak yang berkepentingan yaitu Bapak Ahmad Riyad Firdaus selaku Wakil Direktur 1 Bidang Akademik Politeknik Negeri Batam. Hasil diskusi tersebut menghasilkan beberapa kebutuhan, seperti tidak lagi perlu melakukan ekspor data hasil seleksi UMPB ke Microsoft Excel, menampilkan informasi data calon mahasiswa yang dinyatakan lulus UMPB sesuai keperluan, seperti kelas, gelombang penerimaan, dan tahun ajaran.

#### B. Rencana Cepat (*Quick Plan*)

Pada tahap ini, perencanaan sistem disusun berdasarkan data yang telah didapatkan selama proses komunikasi. Langkah ini akan menghasilkan dokumen kebutuhan pengguna yang mencerminkan seluruh kebutuhan perusahaan dalam pengembangan sistem [11]. Berikut beberapa rancangan fungsional kebutuhan sistem yang didapatkan:

- Sistem dapat berintegrasi dengan Moodle pelaksanaan seleksi UMPB.

- Sistem menghasilkan data kelulusan pendaftar UMPB berdasarkan nilai tertinggi hasil seleksi UMPB untuk tiap program studi.
- Sistem dapat menampilkan data pendaftar yang dinyatakan lulus maupun tidak lulus.

#### C. Pemodelan Desain Cepat (*Modeling Quick Design*)

Pada langkah ini, dokumen kebutuhan pengguna dan data yang telah terkumpul kemudian diwujudkan dalam bentuk model atau *mockup* sistem sebelum dipindahkan ke dalam struktur kode (*coding*) saat diimplementasikan. Tahap ini melibatkan perancangan algoritma, struktur data, dan penggunaan *Unified Modeling Language* (UML) [11].

#### D. Pembangunan Prototipe (*Construction of Prototype*)

Pada langkah ini, aplikasi web dibangun sesuai dengan rancangan yang telah dibuat pada tahap sebelumnya. Semua perencanaan dan data terkait sistem yang akan dibuat diterapkan dalam kode program komputer (*coding*). Tahap ini akan menerjemahkan setiap fungsi sistem yang diminta oleh user [12]. Dalam proses pembuatan sistem ini, bahasa pemrograman yang digunakan adalah PHP dan basis data yang digunakan adalah Oracle.

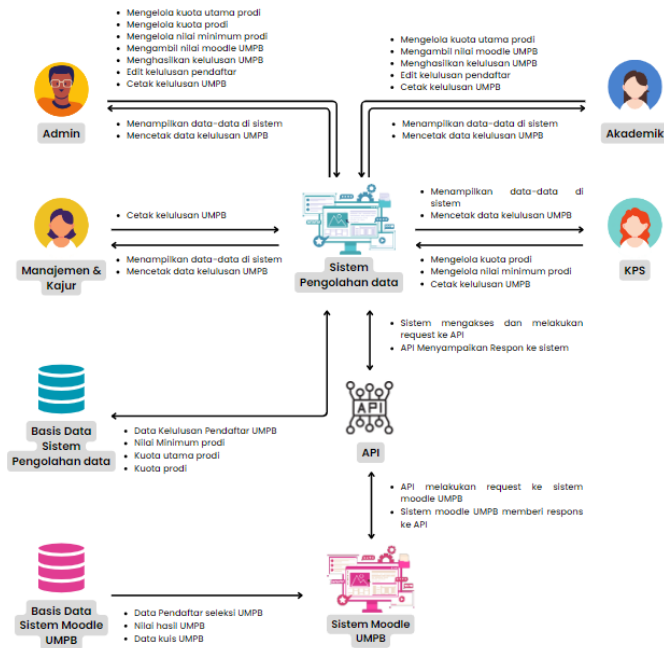
#### E. Pengiriman dan Umpan Balik Penerapan (*Deployment Delivery & Feedback*)

Tahap *Deployment Delivery & Feedback* ini dapat dianggap sebagai tahap akhir dalam proses pembuatan sistem. Sistem yang telah dibangun akan dilakukan beberapa pengujian sebelum diserahkan kepada pengguna. Jika tidak memenuhi kebutuhan pengguna, langkah selanjutnya adalah melakukan identifikasi ulang terhadap kebutuhan pengguna. Kemudian akan direvisi sesuai dengan masukan yang diberikan [13]. Metode pengujian yang digunakan pada pembuatan aplikasi ini adalah Metode *Black Box*.

Tahap ini melibatkan dua kali pertemuan dengan klien yang menghasilkan umpan balik. Pada pertemuan pertama, klien memberikan masukan mengenai sistem, dimana kepala program studi dapat memasukkan nilai minimum dan kuota program studi. Pada pertemuan kedua, masukan diterima mengenai penyempurnaan perhitungan kuota tanpa memerlukan proses *generate* kelulusan terlebih dahulu, dan peningkatan dalam waktu pemuatan nilai UMPB dari Moodle ke dalam *database* yang terlalu lama serta permintaan adanya kelola data kelulusan pendaftar yang dapat mengubah prodi kelulusan pendaftar menjadi tidak lulus. Umpan balik ini penting untuk memastikan bahwa sistem yang dibangun sesuai dengan kebutuhan dan harapan pengguna. Masukan dari klien telah dilakukan dalam implementasi sistem tersebut seperti adanya menu kuota program studi dan menu nilai minimum.

## IV. ANALISIS DAN PERANCANGAN

A. Deskripsi Umum Sistem



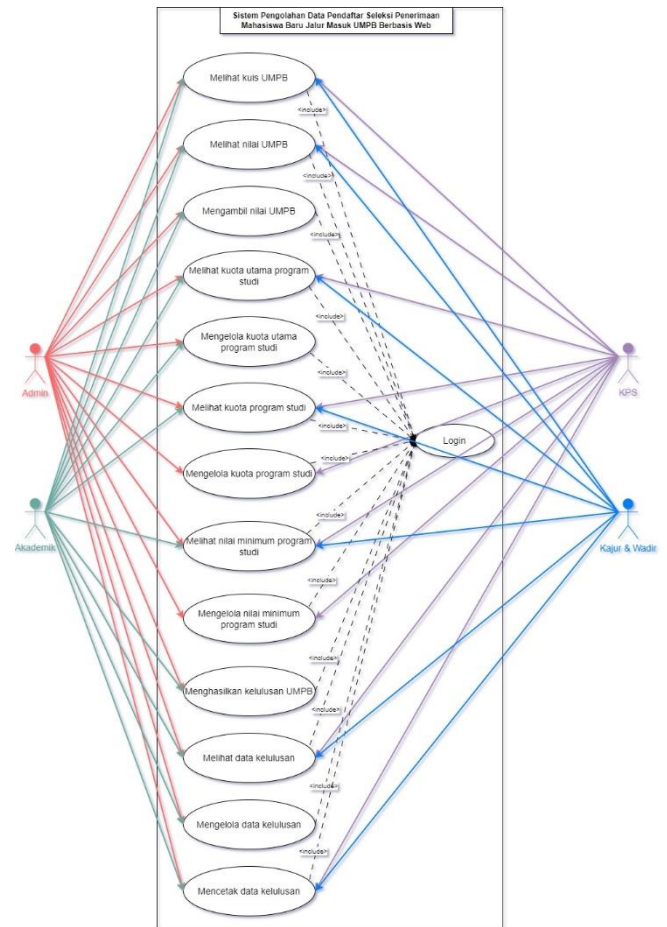
Gambar 2. Deskripsi umum sistem

Gambar 2 merupakan deskripsi umum dari sistem pengolahan data. Dapat dilihat beberapa aktor terlibat yang berinteraksi dengan Sistem Pengolahan Data. Sistem ini terhubung dengan Sistem Moodle UMPB menggunakan API (*Application Programming Interface*), di mana Sistem Pengolahan Data akan mengambil informasi terkait pendaftar, data ujian, dan nilai dari Sistem Moodle UMPB yang berfungsi sebagai platform penyelenggara seleksi UMPB. Sistem Pengolahan Data memiliki basis data sebagai tempat penyimpanan untuk hasil pemrosesan nilai minimum, kuota utama, kuota prodi dan data kelulusan UMPB.

B. Perancangan Diagram

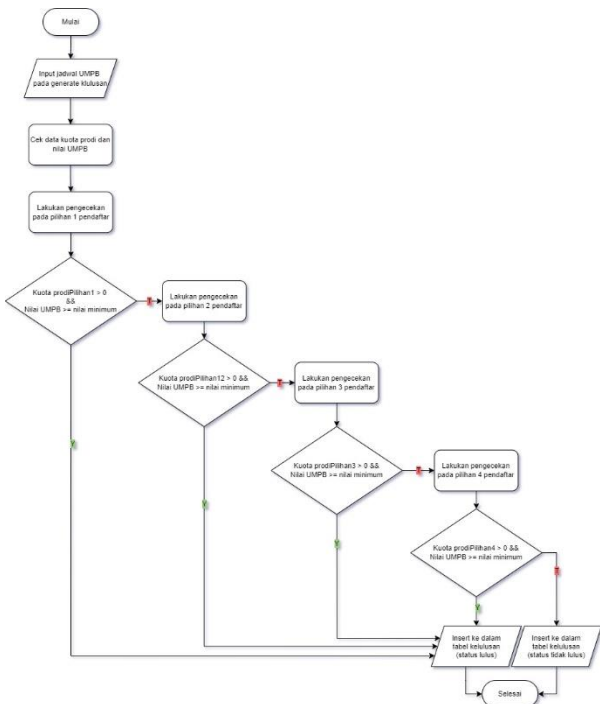
1) *Use Case Diagram*: *Use case diagram* yang terdapat pada Gambar 3 dapat dilihat otorisasi dari setiap aktor, pertama yaitu Admin yang memiliki kewenangan dapat mengakses semua fungsi pada sistem. Kedua adalah Akademik, aktor tersebut memiliki hampir seluruh hak akses pada sistem kecuali mengelola kuota program studi dan mengelola nilai minimum program studi. Ketiga adalah KPS, dapat melakukan *login*, lihat ujian UMPB, lihat nilai UMPB, lihat kuota utama program studi, lihat kuota program studi, mengelola kuota program studi, lihat nilai minimum program studi, mengelola nilai minimum program studi, melihat data kelulusan dan mencetak data kelulusan. Keempat yaitu Kajar (Ketua jurusan) dan Wadir (Wakil direktur) yang hanya memiliki akses lihat data seperti lihat ujian UMPB, lihat nilai UMPB, lihat kuota utama program studi, lihat kuota program studi, lihat nilai minimum, lihat data kelulusan dan mencetak data kelulusan. Untuk aktor KPS, dalam melakukan

pengelolaan kuota prodi dan nilai minimum hanya dapat mengelola bagian prodi yang terlibat saja, tidak bisa mengelola program studi yang bukan bagiannya.



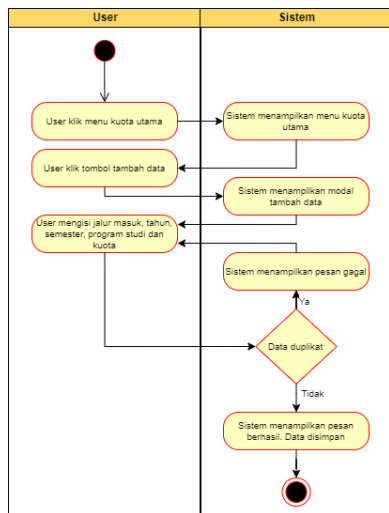
Gambar 3. Use case diagram

2) *Flowchart Pengolahan Nilai UMPB*: *Flowchart* pada Gambar 4 pengolahan nilai dimulai jika pengguna melakukan *generate* kelulusan dengan memasukkan jadwal UMPB kemudian sistem akan memeriksa ketersediaan nilai hasil UMPB dan kuota program studi. Seluruh nilai pendaftar kemudian disusun berdasarkan nilai tertinggi kedalam setiap program studi pilihan pertama. Apabila kuota prodi dan nilai pendaftar mencukupi untuk masuk ke program studi pilihan pertama, nilai beserta data pendaftar akan dimasukkan ke dalam basis data tabel kelulusan dengan status lulus. Namun, jika kuota prodi dan nilai pendaftar tidak mencukupi untuk masuk ke program studi pilihan pertama, sistem akan memeriksa pada program studi pilihan kedua, dan seterusnya. Jika pada akhirnya nilai pendaftar tidak memenuhi kuota penerimaan hingga program studi pilihan keempat, maka pendaftar dianggap tidak lulus dalam seleksi, dan data serta nilai pendaftar tersebut akan dimasukkan ke basis data tabel kelulusan dengan status "tidak lulus".



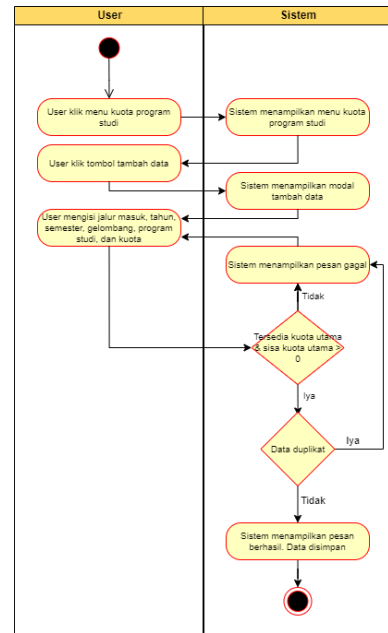
Gambar 4. Flowchart pengolahan nilai UMPB

3) Activity Diagram



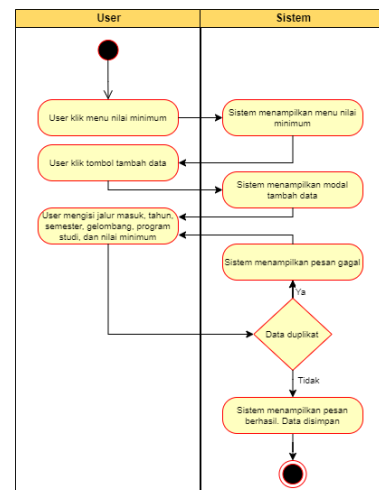
Gambar 5. Diagram aktivitas input kuota utama

Diagram pada Gambar 5 menggambarkan cara proses input kuota utama. Proses tersebut dilakukan oleh aktor Akademik melalui menu kuota utama. Dengan memasukkan informasi seperti jalur masuk, tahun, semester, program studi, dan kuota. Sistem kemudian memeriksa apakah data yang dimasukkan oleh Akademik sudah ada dalam sistem. Jika sudah ada, Akademik harus memasukkan data yang berbeda yang belum ada dalam sistem. Namun, jika belum tersedia, data akan disimpan dan sistem akan memberikan pesan berhasil kepada pengguna.



Gambar 6. Diagram aktivitas input kuota program studi

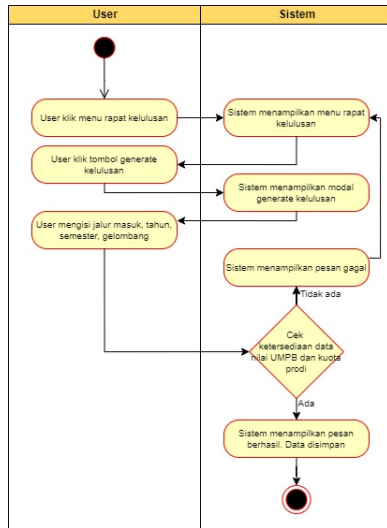
Diagram pada Gambar 6 menggambarkan cara proses input kuota program studi. Proses tersebut dilakukan oleh aktor KPS melalui menu kuota program studi. Dengan memasukkan informasi seperti jalur masuk, tahun, semester, gelombang, program studi, dan kuota. Sistem kemudian memeriksa apakah data yang dimasukkan oleh KPS tersedia kuota utamanya dan sisa kuota lebih besar dari 0. Jika tidak, maka sistem akan menampilkan pesan gagal. Jika ada, maka sistem akan memeriksa data tersebut sudah tersedia di sistem atau tidak. Apabila sudah tersedia, KPS harus memasukkan data yang berbeda yang belum ada dalam sistem. Namun, jika belum tersedia, data akan disimpan dan sistem akan memberikan pesan berhasil kepada pengguna.



Gambar 7. Diagram aktivitas input nilai minimum

Diagram pada Gambar 7 menggambarkan cara proses input nilai minimum. Proses tersebut dilakukan oleh aktor KPS

melalui menu nilai minimum. Dengan memasukkan informasi seperti jalur masuk, tahun, semester, gelombang, program studi, dan nilai minimum. Sistem kemudian memeriksa apakah data yang dimasukkan oleh KPS sudah ada dalam sistem. Jika sudah ada, KPS harus memasukkan data yang berbeda yang belum ada dalam sistem. Namun, jika belum tersedia, data akan disimpan dan sistem akan memberikan pesan berhasil kepada pengguna.

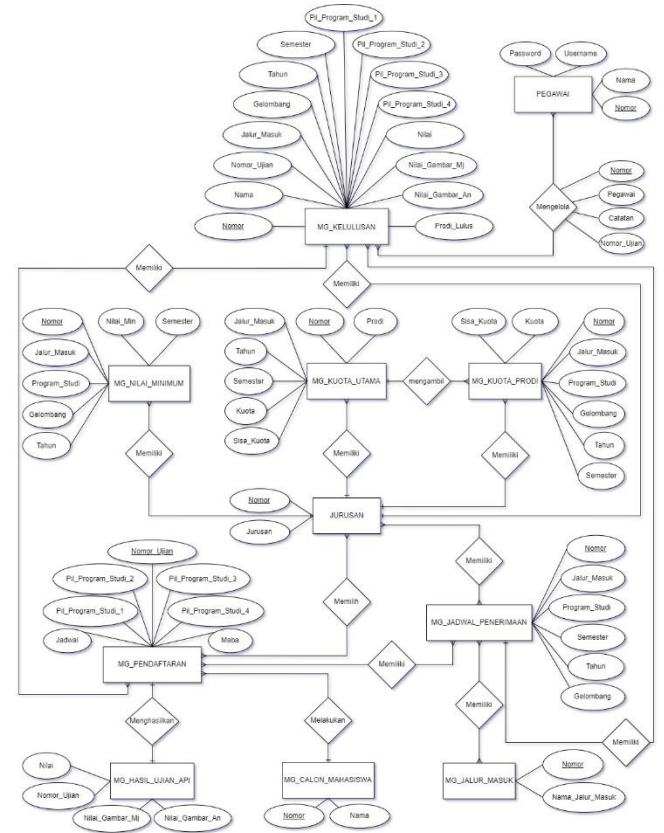


Gambar 8. Diagram aktivitas input nilai minimum

Diagram pada Gambar 8 menggambarkan cara *generate* kelulusan untuk menghasilkan data kelulusan. Proses tersebut dilakukan oleh aktor Akademik melalui menu rapat kelulusan. Dengan memasukkan informasi seperti jalur masuk, tahun, semester, gelombang. Sistem kemudian memeriksa apakah di *database* terdapat nilai hasil UMPB dan kuota program studi sesuai jadwal yang diminta. Jika sudah ada, sistem akan memberikan pesan berhasil. Namun, jika belum tersedia, sistem akan memberikan pesan gagal kepada pengguna.

4) *ER Diagram*: *ER Diagram* sistem yang dirancang pada Gambar 9 memiliki 11 tabel utama yaitu tabel MG\_KELULUSAN, PEGAWAI, MG\_NILAI\_MINIMUM, MG\_KUOTA\_UTAMA, MG\_JADWAL\_PENERIMAAN, MG\_KUOTA\_PRODI, JURUSAN, MG\_PENDAFTARAN, MG\_HASIL\_KUIS\_API, MG\_CALON\_MAHASISWA, dan MG\_JALUR\_MASUK. Masing-masing tabel sudah didefinisikan atributnya serta relasi antar tabel. Dimana calon mahasiswa dapat melakukan pendaftaran lebih dari 1 kali. Setiap pendaftar memilih 4 jurusan (prodi) yang menjadi pilihannya, yang dimana setiap prodi memiliki nilai minimum dan kuota program studi yang diambil dari kuota utama untuk menentukan kelulusan penerimaan mahasiswa baru. Setiap pendaftar memiliki jadwal penerimaan yang terdapat jalur masuk dan beberapa prodi yang dibuka. Data kelulusan yang sudah didapatkan akan disimpan pada tabel kelulusan. Data

kelulusan dapat di kelola oleh pegawai dan akan terekam di database pada tabel MG\_CATATAN.

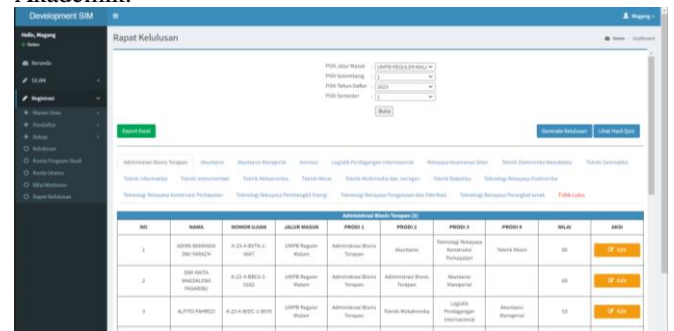


Gambar 9. ER Diagram

## V. HASIL DAN PEMBAHASAN

### A. Implementasi Sistem

Gambar 10 merupakan tampilan dari hasil kelulusan UMPB berdasarkan kuota utama, kuota prodi dan nilai minimum yang telah ditentukan sebelumnya. Halaman ini dapat diakses oleh semua aktor, namun untuk “Generate kelulusan” dan “Edit” hanya dapat dilakukan oleh Admin dan Akademik.



Gambar 10. Implementasi sistem

## B. Pengujian Sistem

Pengujian sistem dilakukan dengan Metode *Black Box*. Metode *Black Box* adalah sebuah pendekatan yang melibatkan pembuatan kasus uji yang mencakup pengujian semua fungsi dalam suatu aplikasi tanpa memperhatikan detail implementasi internalnya. Kasus uji tersebut dirancang

untuk menguji kotak hitam dengan menguji berbagai kemungkinan input untuk melihat apakah aplikasi tersebut memberikan *output* yang benar atau salah [12]. Berikut hasil pengujian sistem yang dilakukan menggunakan Metode *Black Box* terdapat pada Tabel I.

TABEL I  
TABEL PENGUJIAN SISTEM

No	Fitur	Skenario	Kondisi	Yang Diharapkan	Hasil Uji	Kesimpulan
1.	Lihat ujian	Pada menu rapat kelulusan, klik tombol "Lihat Hasil Ujian"		Sistem dapat terhubung dengan API Moodle Learning Registrasi Polibatam dan menampilkan data course dan ujian UMPB	Sistem terhubung dengan API Moodle Learning Registrasi Polibatam dan menampilkan data course dan ujian UMPB	Sukses
2.	Lihat nilai UMPB	Klik tombol "Lihat Nilai" pada tabel ujian		Sistem dapat terhubung dengan API Moodle Learning Registrasi Polibatam dan menampilkan nilai hasil UMPB	Sistem terhubung dengan API Moodle Learning Registrasi Polibatam dan menampilkan nilai hasil UMPB	Sukses
3.	Simpan nilai UMPB (per ujian)	Klik tombol "Simpan Nilai" pada halaman lihat nilai		Sistem menyimpan data kedalam database berdasarkan konten module dan menampilkan alert "Berhasil simpan data"	Sistem menyimpan data kedalam database berdasarkan konten module dan menampilkan alert "Berhasil simpan data"	Sukses
4.	Simpan nilai UMPB (menggunakan rentang waktu)	Pada halaman lihat ujian, mengisi form tanggal "Dari Tanggal" dan "Sampai Tanggal" kemudian klik tombol "Arsip data"	Terdapat ujian yang dibuka pada rentang waktu yang telah ditentukan	Sistem menyimpan data ke dalam database berdasarkan konten module dan menampilkan alert "Berhasil arsip data"	Sistem menyimpan data kedalam database berdasarkan konten module dan menampilkan alert "Berhasil arsip data"	Sukses
5.	Simpan nilai UMPB (menggunakan rentang waktu)	Pada halaman lihat ujian, mengisi form tanggal "Dari Tanggal" dan "Sampai Tanggal" kemudian klik tombol "Arsip data"	Tidak ada ujian yang dibuka pada rentang waktu yang telah ditentukan	Tidak ada data yang disimpan ke dalam database dan menampilkan alert "Gagal: Tidak ditemukan ujian pada jadwal tersebut"	Tidak ada data yang disimpan ke dalam database dan menampilkan alert "Gagal: Tidak ditemukan ujian pada jadwal tersebut"	Sukses
6.	Mengelola nilai minimum prodi	Pada menu nilai minimum, klik tombol "Tambah data", mengisi form, kemudian klik "Simpan data"		Sistem menyimpan data kedalam database dan menampilkan alert "Berhasil tambah data"	Sistem menyimpan data kedalam database dan menampilkan alert "Berhasil tambah data"	Sukses
7.	Mengelola nilai minimum prodi	Pada menu nilai minimum, klik tombol "Tambah data",	Mengulang tambah data yang sama	Tidak ada data yang disimpan ke database dan menampilkan alert "Gagal : data sudah tersedia"	Tidak ada data yang disimpan ke database dan menampilkan alert "Gagal : data sudah tersedia"	Sukses

		mengisi form, kemudian klik "Simpan data"				
8.	Lihat nilai minimum prodi	Pada menu nilai minimum		Sistem dapat terhubung dengan database dan menampilkan data nilai minimum	Sistem dapat terhubung dengan databased dan menampilkan data nilai minimum	Sukses
9.	Mengelola nilai minimum prodi	Pada menu nilai minimum, klik tombol "Edit", ubah nilai kemudian klik "Simpan data"		Database akan update kuota, dan sistem akan menampilkan alert "Berhasil memperbarui data"	Database akan update kuota, dan sistem akan menampilkan alert "Berhasil memperbarui data"	Sukses
10.	Mengelola nilai minimum prodi	Pada menu nilai minimum, klik tombol "Hapus"		Sistem dapat menghapus data dari database dan menampilkan alert "Berhasil hapus data"	Sistem dapat menghapus data dari database dan menampilkan alert "Berhasil hapus data"	Sukses
11.	Mengelola kuota utama	Pada menu kuota utama klik "Tambah data", mengisi form, kemudian klik "Simpan data"		Sistem menyimpan data kedalam database dan menampilkan alert "Berhasil tambah data"	Sistem menyimpan data kedalam database dan menampilkan alert "Berhasil tambah data"	Sukses
11.	Mengelola kuota utama	Pada menu kuota utama klik "Tambah data", mengisi form, kemudian klik "Simpan data"	Mengulang tambah data	Tidak ada data yang disimpan ke database dan menampilkan alert "Gagal : data sudah tersedia"	Tidak ada data yang disimpan ke database dan menampilkan alert "Gagal : data sudah tersedia"	Sukses
12.	Lihat kuota utama	Pada menu kuota utama		Sistem dapat terhubung dengan database dan menampilkan data kuota utama	Sistem dapat terhubung dengan databased dan menampilkan data kuota utama	Sukses
13.	Mengelola kuota utama	Pada menu kuota utama, klik tombol "Edit", ubah nilai kemudian klik "Simpan data"	Kuota utama >= kuota prodi	Database akan update kuota dan sisa kuota utama, kemudian sistem akan menampilkan alert "Berhasil memperbarui data"	Database akan update kuota dan sisa kuota utama, kemudian sistem akan menampilkan alert "Berhasil memperbarui data"	Sukses
14.	Mengelola kuota utama	Pada menu kuota utama, klik tombol "Edit", ubah nilai kemudian klik "Simpan data"	Kuota utama < kuota prodi	Database gagal update kuota dan sisa kuota, kemudian sistem akan menampilkan alert "Gagal memperbarui data"	Database akan update kuota dan sisa kuota, kemudian sistem akan menampilkan alert "Gagal memperbarui data"	Sukses
15.	Mengelola kuota utama	Pada menu kuota utama, klik tombol "Hapus"	Belum ada data dengan jadwal yang sama di kuota prodi	Sistem dapat menghapus data dari database dan menampilkan alert "Berhasil hapus data"	Sistem dapat menghapus data dari database dan menampilkan alert "Berhasil hapus data"	Sukses
16.	Mengelola kuota utama	Pada menu kuota utama, klik tombol "Hapus"	Sudah ada data dengan jadwal yang sama di kuota prodi	Sistem tidak menghapus data dari database dan menampilkan alert "Gagal hapus data"	Sistem tidak menghapus data dari database dan menampilkan alert "Gagal hapus data"	Sukses
17.	Mengelola kuota prodi	Pada menu kuota prodi klik "Tambah data", mengisi form, kemudian klik "Simpan data"		Sistem menyimpan data kedalam database dan menampilkan alert "Berhasil tambah data"	Sistem menyimpan data kedalam database dan menampilkan alert "Berhasil tambah data"	Sukses

18.	Mengelola kuota prodi	Pada menu kuota prodi klik "Tambah data", mengisi form, kemudian klik "Simpan data"	Mengulang tambah data	Tidak ada data yang disimpan ke database dan menampilkan alert "Gagal : data sudah tersedia"	Tidak ada data yang disimpan ke database dan menampilkan alert "Gagal : data sudah tersedia"	Sukses
19.	Mengelola kuota prodi	Pada menu kuota prodi klik "Tambah data", mengisi form, kemudian klik "Simpan data"	Belum tersedia kuota utamanya	Tidak ada data yang disimpan ke database dan menampilkan alert "Gagal : kuota utama belum tersedia"	Tidak ada data yang disimpan ke database dan menampilkan alert "Gagal : kuota utama belum tersedia"	Sukses
20.	Lihat kuota prodi	Pada menu kuota prodi		Sistem dapat terhubung dengan database dan menampilkan data kuota prodi	Sistem dapat terhubung dengan databased dan menampilkan data kuota prodi	Sukses
	Mengelola kuota prodi	Pada menu kuota prodi, klik tombol "Edit", ubah nilai kemudian klik "Simpan data"	Jika belum <i>generate</i> kelulusan	Database akan update sisa kuota utama dan kuota prodi , kemudian sistem akan menampilkan alert "Berhasil memperbarui data"	Database akan update sisa kuota utama, kuota prodi , kemudian sistem akan menampilkan alert "Berhasil memperbarui data"	Sukses
21.	Mengelola kuota prodi	Pada menu kuota prodi, klik tombol "Edit", ubah nilai kemudian klik "Simpan data"	Jika sudah <i>generate</i> kelulusan, kuota $\geq$ jumlah data lulus	Database akan update sisa kuota utama, kuota prodi dan sisa kuota prodi, kemudian sistem akan menampilkan alert "Berhasil memperbarui data"	Database akan update sisa kuota utama, kuota prodi dan sisa kuota prodi, kemudian sistem akan menampilkan alert "Berhasil memperbarui data"	Sukses
22.	Mengelola kuota prodi	Pada menu kuota prodi, klik tombol "Edit", ubah nilai kemudian klik "Simpan data"	Jika sudah <i>generate</i> kelulusan, kuota $<$ jumlah data lulus	Tidak ada data yang ke update di database, sistem akan menampilkan alert "Gagal memperbarui data"	Tidak ada data yang ke update di database, sistem akan menampilkan alert "Gagal memperbarui data"	Sukses
23.	Mengelola kuota prodi	Pada menu kuota prodi, klik tombol "Hapus"		Sistem dapat menghapus data dari database dan menampilkan alert "Berhasil hapus data"	Sistem dapat menghapus data dari database dan menampilkan alert "Berhasil hapus data"	Sukses
24.	Menghasilkan kelulusan UMPB	Klik tombol "Generate Kelulusan" kemudian mengisi form jalur masuk, gelombang, tahun daftar dan semester kemudian klik tombol "Simpan"	Data yang diinput tersedia	Sistem akan mengelola data dan nilai pendaftar kemudian menampilkan alert "Berhasil <i>generate</i> kelulusan"	Sistem akan mengelola data dan nilai pendaftar kemudian menampilkan alert "Berhasil <i>generate</i> kelulusan"	Sukses
25.	Menghasilkan kelulusan UMPB	Klik tombol "Generate Kelulusan" mengisi form jalur masuk, gelombang, tahun daftar dan semester lalu klik tombol "Simpan"	Data tidak tersedia, belum simpan nilai dari Moodle learning registrasi Polibatam	Sistem menampilkan alert "Gagal <i>generate</i> kelulusan, data nilai UMPB belum ditambahkan"	Sistem menampilkan alert "Gagal <i>generate</i> kelulusan, data nilai UMPB belum ditambahkan"	Sukses

26.	Lihat data kelulusan	Mengisi multifilter jalur masuk, gelombang, tahun daftar dan semester kemudian klik tombol "Buka"	Data yang dimasukkan tersedia	Sistem akan menampilkan tab nama program studi yang berisi tabel data kelulusan pendaftar	Sistem akan menampilkan tab nama program studi yang berisi tabel data kelulusan pendaftar	Sukses
27.	Lihat data kelulusan	Mengisi multifilter kemudian klik tombol "Buka"	Data yang dimasukkan tidak tersedia	Sistem akan menampilkan alert "Belum dilakukan <i>generate</i> kelulusan pada jalur masuk, gelombang, semester atau tahun"	Sistem akan menampilkan alert "Belum dilakukan <i>generate</i> kelulusan pada jalur masuk, gelombang, semester atau tahun"	Sukses
28.	Mengelola data kelulusan	Klik tombol "Edit", pilih tujuan prodi lulus kemudian klik tombol "Simpan data"	Ubah prodi lulus menjadi tidak lulus	Sistem akan mengupdate prodi lulus di database menjadi 0 (tidak lulus) dan sisa kuota prodi lama akan bertambah 1, kemudian sistem akan menampilkan alert "Berhasil edit data"	Sistem akan mengupdate prodi lulus di database menjadi 0 (tidak lulus) dan sisa kuota prodi lama akan bertambah 1, kemudian sistem akan menampilkan alert "Berhasil edit data"	Sukses
29.	Mengelola data kelulusan	Klik tombol "Edit", pilih tujuan prodi lulus kemudian klik tombol "Simpan data"	Ubah prodi lulus ke prodi lain, dengan prodi tujuan memiliki sisa kuota prodi = 0	Sistem akan menampilkan alert "Gagal edit data, kuota program studi sudah habis"	Sistem akan menampilkan alert "Gagal edit data, kuota program studi sudah habis"	Sukses
30.	Mengelola data kelulusan	Klik tombol "Edit", pilih tujuan prodi lulus kemudian klik tombol "Simpan data"	Ubah prodi lulus ke prodi lain, dengan prodi tujuan memiliki sisa kuota prodi > 0	Sistem akan mengupdate prodi lulus, kemudian sisa kuota prodi tujuan - 1, sisa kuota prodi lama + 1, kemudian menampilkan alert "Berhasil edit data"	Sistem akan mengupdate prodi lulus, kemudian sisa kuota prodi tujuan - 1, sisa kuota prodi lama + 1, kemudian menampilkan alert "Berhasil edit data"	Sukses
31.	Mengelola data kelulusan	Klik tombol "Edit", pilih tujuan prodi lulus kemudian klik tombol "Simpan data"	Ubah prodi lulus dengan status tidak lulus ke prodi yang dibuka, dengan prodi tujuan memiliki sisa kuota prodi = 0	Sistem akan menampilkan alert "Gagal edit data, kuota program studi sudah habis"	Sistem akan menampilkan alert "Gagal edit data, kuota program studi sudah habis"	Sukses
32.	Mengelola data kelulusan	Klik tombol "Edit", pilih tujuan prodi lulus kemudian klik tombol "Simpan data"	Ubah prodi lulus dengan status tidak lulus ke prodi yang dibuka, dengan prodi tujuan memiliki sisa kuota prodi > 0	Sistem akan mengupdate prodi lulus, kemudian sisa kuota utama tujuan - 1, kemudian menampilkan alert "Berhasil edit data"	Sistem akan mengupdate prodi lulus, kemudian kuota prodi tujuan + 1, sisa kuota utama tujuan - 1, kemudian menampilkan alert "Berhasil edit data"	Sukses
33.	Cetak data kelulusan	Pada menu rapat kelulusan isi multi filter kemudian klik "cetak data"		Sistem akan otomatis mengunduh file excel dengan format nama "jalur masuk _gelombang_semester_tahun .xls"	Sistem akan otomatis mengunduh file excel dengan format nama "jalur masuk _gelombang_semester_tahun .xls"	

## VI. KESIMPULAN

Dari hasil perancangan sistem pengolahan data pendaftar seleksi penerimaan mahasiswa baru melalui jalur masuk UMPB berbasis web, dapat disimpulkan bahwa sistem ini telah berhasil berintegrasi dengan Moodle UMPB melalui API. Dengan integrasi ini, sistem dapat mengakses data ujian dan nilai dari pelaksanaan UMPB dengan mudah, memudahkan pengambilan nilai yang diperlukan untuk pengolahan data pendaftar. Selain itu, sistem telah berhasil mengolah data dan nilai pendaftar dengan menggunakan algoritma yang telah diprogram dengan baik. Hal ini membuat sistem menjalankan proses seleksi secara otomatis. Dengan adanya sistem pengolahan data yang terkomputerisasi, kesalahan dalam penginputan data dapat diminimalkan. Hal ini dapat meningkatkan akurasi data dan mengurangi potensi kesalahan dalam proses seleksi. Sistem ini juga menghilangkan ketergantungan pada Microsoft Excel dalam proses rapat kelulusan untuk menentukan kelulusan setiap pendaftar UMPB. Selain itu, sistem juga memberikan transparansi terhadap kuota prodi, yang dapat membantu dalam menghindari potensi *overload* penerimaan mahasiswa baru.

## VII. SARAN

Untuk langkah pengembangan berikutnya, diharapkan sistem dapat diperluas menjadi lebih kompleks dengan penambahan fitur atau menu yang dapat meningkatkan fungsionalitasnya. Integrasi dengan berbagai sistem internal yang ada, sebagai langkah penting untuk menciptakan kesatuan sistem yang lebih besar dan lebih signifikan. Dengan demikian, pengguna akan dapat menikmati pengalaman yang lebih lengkap dan terintegrasi, serta memungkinkan sistem ini untuk lebih responsif terhadap kebutuhan pengguna secara keseluruhan.

## DAFTAR PUSTAKA

- [1] Polibatam. (2023). UMPB (Ujian Masuk Polibatam). <https://registrasi.polibatam.ac.id/statistik/>.
- [2] Rohmat, T., & Pertiwi, D. D. (2020). Analisis dan Desain Sistem Informasi Pengolahan Nilai Siswa di SMK Avicena Rajeg. *JIKA (Jurnal Informatika)*, 4(1), 29. <https://doi.org/10.31000/jika.v4i1.2571>.
- [3] Widiawati, & Siradjuddin, H. K. (2020). Perancangan Website Sistem Seleksi Siswa Baru menggunakan Framework CodeIgniter Pada Madrasah Aliyah Alkhairaat Kalumpang Kota Ternate. *Jurnal Ilmiah ILKOMINFO - Ilmu Komputer & Informatika*, 3(2). <https://doi.org/10.47324/ilkominfo.v3i2.105>.
- [4] Irawan, Y., & Wahyuni, R. (2019). Sistem Pendukung Keputusan Penerimaan Siswa Baru di SMK Negeri 1 Tapung Hulu Menggunakan Metode Simple Multi Attribut Rating Technique (SMART). *JOISIE (Journal Of Information Systems And Informatics Engineering)*, 3(1), 25. <https://doi.org/10.35145/joisie.v3i1.405>.
- [5] Gunawan, R., Yudianta, Y., & Apriansyah, W. Y. (2021). Rancang Bangun Company Profile Kebab Ben's Berbasis Web Menggunakan Framework Codeigniter. *Dirgamaya: Jurnal Manajemen Dan Sistem Informasi*, 1(2), 36–45. <https://doi.org/10.35969/dirgamaya.v1i2.181>.
- [6] Polibatam. (2023). UMPB (Ujian Masuk Polibatam). <https://registrasi.polibatam.ac.id/jalur-masuk/umpb/>.
- [7] Herlalang, E., & Kautsar, I. A. (2022). Rancang Bangun Fitur Chat Pada Eportofolio Berbasis Web. *JUPI (Jurnal Ilmiah Penelitian Dan Pembelajaran Informatika)*, 7(3), 780–791. <https://doi.org/10.29100/jupi.v7i3.2990>.
- [8] Heripracoyo, S. (2018). Penggunaan Database Oracle Di Beberapa Perusahaan Indonesia. Retrieved from <https://sis.binus.ac.id/2018/08/08/penggunaan-database-oracle-dibeberapa-perusahaan-indonesia/>.
- [9] Carr, C. N., Silva, E. de V., Lima, G. C. O., Carr, R. F., & Braga, T. R. da C. (2023). Oracle Database: An In-Depth Analysis of the DBMS That Dominates the Corporate World. *Research, Society and Development*, 12(5), e0812541323. <https://doi.org/10.33448/rsd-v12i5.41323>.
- [10] Nurhadi, & Muhammad Ridwan. (2022). Sistem Informasi Inventaris Berbasis Web Menggunakan Metode Prototype. *Jurnal Multidisiplin Madani*, 2(9), 3543–3550. <https://doi.org/10.55927/mudima.v2i9.1143>.
- [11] Kustanto, G. E. A., & Chernovita, H. P. (2021). Perancangan Sistem Informasi Manajemen Berbasis Web Studi Kasus: PT Unicorn Intertranz. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 8(4), 719. <https://doi.org/10.25126/jtiik.2021844849>.
- [12] Aditya, R., Pranatawijaya, V. H., & Putra, P. B. A. A. (2021). Rancang Bangun Aplikasi Monitoring Kegiatan Menggunakan Metode Prototype. *Journal of Information Technology and Computer Science*, 1(1), 47-57.
- [13] Hasan, Y. A., Mardiana, M., & Nama, G. F. (2022). Sistem Pendeteksi Kebocoran Tabung Gas Lpg Otomatis Berbasis Arduino Uno Menggunakan Metode Prototype. *Jurnal Informatika Dan Teknik Elektro Terapan*, 10(3). <https://doi.org/10.23960/jitet.v10i3.2671>.