



KERJA PRAKTIK - IF184801

Rancang Bangun Aplikasi Manajemen dan Pendaftaran Peserta Didik Baru Kabupaten Bojonegoro Tahun Ajar 2020/2021

Dinas Pendidikan Kabupaten Bojonegoro

Jl. Panglima Polim No.36, Sumbang, Kec. Bojonegoro,
Kabupaten Bojonegoro, Jawa Timur 62115

Periode: 2 Mei 2020 - 8 Agustus 2020

Oleh:

Akwila Feliciano P 05111740000181

Setya Wibawa 05111740000028

Pembimbing Jurusan

Dr. Eng. RADITYO ANGGORO, S.Kom., M.Sc.

Pembimbing Lapangan

Faizal Johan Atletiko, S.Kom., MT.

DEPARTEMEN TEKNIK INFORMATIKA

Fakultas Teknologi Elektro dan Informatika Cerdas

Institut Teknologi Sepuluh Nopember

Surabaya 2020

[Halaman ini sengaja dikosongkan]

LEMBAR PENGESAHAN

KERJA PRAKTIK

Rancang Bangun Aplikasi Manajemen dan Pendaftaran Peserta Didik Baru Kabupaten Bojonegoro Tahun Ajar 2020/2021

Oleh:

Akwila Feliciano P

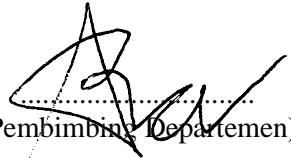
05111740000181

Setya Wibawa

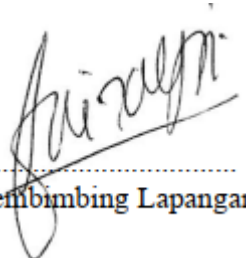
05111740000028

Disetujui oleh Pembimbing Kerja Praktik:

1. Dr. Eng. Radityo Anggoro, S.Kom.,
M.Sc.
NIP. 19841016 2008121002


.....
(Pembimbing Departemen)

2. Faizal Johan Atletiko, S.Kom., MT.
NIP. 197704082002121001


.....
(Pembimbing Lapangan)

SURABAYA
Januari, 2021

[Halaman ini sengaja dikosongkan]

**Rancang Bangun Aplikasi Manajemen dan Pendaftaran
Peserta Didik Baru Kabupaten Bojonegoro
Tahun Ajar 2020/2021**

Nama Mahasiswa : Akwila Feliciano
Pradiptatmaka
NRP : 05111740000181
Nama Mahasiswa : Setya Wibawa
NRP : 05111740000028
Departemen : Teknik Informatika
Fakultas : Fakultas Teknologi Elektro
dan Informatika Cerdas
Pembimbing Jurusan : Dr. Eng. Radityo Anggoro,
S.Kom., M.Sc.

ABSTRAKSI

Aplikasi PPDB SMP Kabupaten Bojonegoro, Jawa Timur merupakan sistem informasi yang memiliki tujuan untuk menjalankan program Dinas Pendidikan Kabupaten Bojonegoro, yaitu penerimaan siswa baru jenjang sekolah menengah pertama secara daring (*online*) dan terintegrasi langsung yang memudahkan *monitoring* dan manajemen secara terpusat.

Dengan melihat kondisi nasional dari adanya pandemi ini, tujuan dari dibentuknya aplikasi sistem informasi ini menjadi sebuah solusi yang praktis dengan memanfaatkan teknologi informasi yang sudah berkembang pesat di era digitalisasi sekarang.

Kata kunci: *penerimaan siswa, sekolah menengah pertama, pandemi, teknologi informasi, aplikasi manajemen*

[Halaman ini sengaja dikosongkan]

KATA PENGANTAR

Puji dan syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas berkat dan limpah rahmat yang diberikan sehingga penulis dapat melaksanakan kegiatan proyek untuk menjadikannya salah satu kewajiban sebagai mahasiswa Teknik Informatika, yaitu Kerja Praktik (KP).

Penulis berharap melalui laporan dari kerja praktik ini dapat menambah wawasan pembaca dan dapat menjadikannya bahan referensi untuk melaksanakan kegiatan serupa sekaligus sebagai bahan evaluasi untuk ke depannya. Penulis sadar bahwa apa yang dituliskan ke dalam laporan kerja praktik ini jauh dari kata sempurna, oleh karena itu penulis mengharapkan saran ataupun kritik yang membangun untuk menjadikan kegiatan serupa menjadi lebih baik ke depannya.

Melalui laporan kerja praktik ini, penulis ingin menyampaikan rasa syukur dan terima kasih kepada pihak-pihak yang telah membantu dalam menjalankan kerja praktik yang penulis lakukan baik secara langsung maupun tidak langsung. Pihak-pihak tersebut antara lain adalah:

1. Kedua orang tua dan keluarga yang mendukung secara langsung dan tidak langsung, baik secara motivasi maupun doa/harapan kepada penulis.
2. Bapak Faizal Johan Atletiko, selaku *leader* dan pembimbing lapangan selama kerja praktik yang telah memberikan dukungan dan rasa percaya kepada penulis.
3. Bapak Dr. Eng. Radityo Anggoro, S.Kom., M.Sc., selaku pembimbing kerja praktik yang telah membimbing penulis hingga penulisan buku ini selesai.

Surabaya, Januari 2021

Penulis

[Halaman ini sengaja dikosongkan]

DAFTAR ISI

LEMBAR PENGESAHAN.....	iii
ABSTRAKSI.....	v
KATA PENGANTAR.....	vii
DAFTAR ISI.....	ix
DAFTAR ISI GAMBAR.....	x
DAFTAR ISI TABEL.....	xi
BAB 1 PENDAHULUAN.....	1
1.1. Latar Belakang.....	1
1.2. Maksud dan Tujuan.....	2
1.3. Manfaat.....	2
1.4. Sasaran.....	3
1.5. Ruang Lingkup.....	4
1.6. Rumusan Masalah.....	5
2.1. Ikhtisar Dokumen.....	6
BAB 2 DESKRIPSI UMUM APLIKASI.....	8
2.1. Deskripsi Umum Sistem.....	8
2.2. Karakteristik Pengguna.....	12
2.3. Lingkungan Operasi.....	14
2.4. Batasan Perancangan dan Implementasi Sistem.....	16
BAB 3 PROSES BISNIS APLIKASI.....	17
3.1. Proses 1.....	17
3.2. Proses 2.....	18
3.3. Proses 3.....	19

3.4	Proses 4	19
BAB 4 ANALISIS DAN DESAIN MODEL		20
4.1	Data Model Aplikasi PPDB SMP Kabupaten Bojonegoro 2020	20
BAB 5 IMPLEMENTASI SISTEM.....		24
5.1.	Implementasi Antarmuka	24
5.1.1.	Aplikasi Manajemen.....	24
5.2.	Implementasi Model Bisnis.....	59
5.2.1.	Aplikasi Manajemen.....	59
5.2.2.	Aplikasi Pendaftaran	104
BAB 6 KESIMPULAN DAN SARAN		120
6.1.	Kesimpulan.....	120
6.2.	Saran.....	121
BIODATA PENULIS.....		124

DAFTAR ISI GAMBAR

Gambar 4.1-1 Diagram Model Database Aplikasi PPDB Kab. Bojonegoro 2020	21
---	----

DAFTAR ISI TABEL

Tabel 2-1 Karakteristik pengguna aplikasi.....	12
Tabel 5.1-1 Kode UI Dashboard Aplikasi Manajemen	25
Tabel 5.1-2 Kode UI Daftar Ulang.....	35
Tabel 5.1-3 Kode UI Lihat Berkas untuk akun administrator	51
Tabel 5.2-1 Kode Model Autentifikasi.....	59
Tabel 5.2-2 Model Pendaftaran pada Aplikasi Manajemen	63
Tabel 5.2-3 Model Siswa mengenai informasi pendaftaran dan pendataan.....	85
Tabel 5.2-4 Model Daftar Ulang Peserta yang diterima.....	95
Tabel 5.2-5 Tabel Autentikasi Aplikasi Pendaftaran	104
Tabel 5.2-6 Model Daftar Ulang Aplikasi Pendaftaran.....	105
Tabel 5.2-7 Model Pendaftaran	106
Tabel 5.2-8 Model Pendaftaran Jalur Prestasi	107
Tabel 5.2-9 Model Peserta Pendaftaran.....	108
Tabel 5.2-10 Model Pilihan Sekolah.....	111
Tabel 5.2-11 Model Prestasi.....	113
Tabel 5.2-12 Model Sekolah	116
Tabel 5.2-13 Model Zona Sekolah	117

[Halaman ini sengaja dikosongkan]

BAB 1

PENDAHULUAN

1.1. Latar Belakang

Proses penerimaan siswa didik baru merupakan salah satu langkah awal dalam proses akademik jenjang sekolah dasar dan sekolah menengah. Pada tahun ajaran 2020/2021, seluruh dunia khususnya Indonesia mengalami pandemi Covid-19, yang mendesak segala kegiatan diselenggarakan secara daring (*online*). Dengan adanya tuntutan aktivitas daring dengan tetap menjalankan program pendidikan seperti biasanya, banyak pihak sekolah yang menggunakan fasilitas penerimaan siswa didik baru secara daring melalui *website*. Untuk mempermudah proses akademik calon siswa didik baru jenjang sekolah menengah pertama (SMP), Dinas Pendidikan Kabupaten Bojonegoro, Jawa Timur menjadikan kesempatan ini untuk membuat sistem terpusat yang memantau proses penerimaan didik baru jenjang SMP dan tiap sekolah tujuan tetap dapat menjalankan penerimaan siswa baru di daerah Kabupaten Bojonegoro. Sistem aplikasi terdiri dari Aplikasi Manajemen Penerimaan Peserta Didik Baru yang digunakan oleh pihak sekolah dan pihak Dinas

Pendidikan Kabupaten Bojonegoro. Sedangkan untuk aplikasi yang digunakan calon siswa untuk mendaftar sekolah SMP menggunakan Aplikasi Penerimaan Peserta Didik Baru Kabupaten Bojonegoro.

1.2. Maksud dan Tujuan

Dokumen ini berisi spesifikasi Aplikasi Penerimaan Peserta Didik Baru (PPDB) Kabupaten Bojonegoro, Jawa Timur tahun ajar 2020/2021. Maksud dari pembuatan sistem Aplikasi PPDB ini adalah untuk menerima pendaftaran calon peserta didik, mengatur, serta *monitoring* pendaftaran oleh pihak sekolah menengah atas Kabupaten Bojonegoro dan pihak Dinas Pendidikan Kabupaten Bojonegoro.

1.3. Manfaat

Dari pengerjaan kerja praktik untuk membangun Aplikasi Penerimaan Peserta Didik Baru (PPDB) Kabupaten Bojonegoro, Jawa Timur tahun ajar 2020/2021, beberapa manfaat yang penulis dapatkan, yaitu:

- Dapat menciptakan sistem aplikasi yang dapat digunakan oleh masyarakat

- Dapat meningkatkan kemampuan *soft skill* dan *hard skill* dari kerjasama dengan tim dan komunikasi, terlebih semasa pandemi dilaksanakan secara penuh daring (*online*).

1.4. Sasaran

Sasaran dari pekerjaan ini yaitu menyelesaikan Aplikasi Penerimaan Peserta Didik Baru yang digunakan oleh pihak sekolah untuk menerima calon didik pada masing-masing sekolah dan pihak dinas pendidikan untuk mendapatkan data-data penerimaan siswa jenjang SMP Kabupaten Bojonegoro serta mengawasi program PPDB secara daring. Sedangkan sasaran dari dokumen ini adalah akan digunakan sebagai bahan acuan dalam proses pengembangan dan sebagai bahan evaluasi pada saat proses pengembangan perangkat lunak maupun di akhir pengembangannya. Dengan adanya dokumen ini diharapkan pengembangan perangkat lunak untuk sistem PPDB akan lebih terarah dan lebih terfokus serta tidak menimbulkan ambiguitas terutama bagi pengembang perangkat lunak sistem informasi.

1.5. Ruang Lingkup

Aplikasi Penerimaan Peserta Didik Baru (PPDB) SMP, Kabupaten Bojonegoro 2020 merupakan sistem aplikasi berbasis *website* untuk menerima pendaftaran, manajemen, dan *monitoring* penerimaan calon peserta didik baru jenjang SMP Negeri di wilayah Kabupaten Bojonegoro, Jawa Timur. Sistem ini dapat memfasilitasi tiga jenis pengguna yang terbagi ke dalam dua sub sistem, yaitu pihak calon peserta didik tingkat SMP, pihak sekolah dan pihak dinas pendidikan setempat sesuai fungsionalitas masing-masing. Dengan adanya Aplikasi Penerimaan Peserta Didik Baru (PPDB) SMP, Kabupaten Bojonegoro 2020 diharapkan kegiatan penerimaan calon didik baru menjadi terfasilitas secara daring (*online*) sekaligus menjadikan langkah ke depan untuk modernisasi teknologi saat ini. Pada Aplikasi Penerimaan Peserta Didik Baru (PPDB) SMP, Kabupaten Bojonegoro 2020 mencakup beberapa fungsi utama, antara lain:

1. Fungsi Penerimaan Data Pendaftaran
2. Fungsi Monitoring Proses Penerimaan
3. Fungsi Manajemen Admin

4. Fungsi Manajemen Unduh Berkas
5. Fungsi Manajemen Data Sekolah
6. Fungsi Monitoring Data Siswa
7. Fungsi Monitoring Jarak Rumah dengan Sekolah
8. Fungsi Manajemen Rangka Calon Siswa
9. Fungsi Pendaftaran Calon Peserta Didik Baru
10. Fungsi Pendataan Calon Peserta Didik Baru

1.6. Rumusan Masalah

1. Bagaimana menyelesaikan permasalahan pendaftaran peserta didik baru secara daring?
2. Bagaimana mencari cara untuk menyeleksi peserta didik baru berdasarkan jarak rumah dengan jarak sekolah?

2.1. Ikhtisar Dokumen

Dokumen ini secara garis besar terdiri dari 6 bab dengan perincian sebagai berikut:

- **Bab 1 Pendahuluan**, merupakan pengantar dokumen yang berisi latar belakang penulisan dokumen, maksud dan tujuan dari pengerjaan aplikasi dan penulisan dokumen, sasaran aplikasi dan dokumen, ruang lingkup pengerjaan aplikasi, serta deskripsi umum dokumen yang merupakan ikhtisar dokumen.
- **Bab 2 Deskripsi Umum Aplikasi**, merupakan bagian yang berisikan deskripsi umum Aplikasi Penerimaan Peserta Didik Baru (PPDB) SMP, Kabupaten Bojonegoro 2020, karakteristik pengguna, lingkungan operasi, serta batasan perancangan dan implementasi sistem aplikasi.
- **Bab 3 Proses Bisnis Aplikasi**, merupakan bagian deskripsi proses-proses bisnis yang dilakukan pada Aplikasi Penerimaan Peserta

Didik Baru (PPDB) SMP, Kabupaten Bojonegoro 2020.

- **Bab 4 Analisis dan Desain Model**, merupakan bagian yang menjelaskan alur penggunaan dan deskripsi model data yang digunakan pada pembuatan sistem Aplikasi Penerimaan Peserta Didik Baru (PPDB) SMP, Kabupaten Bojonegoro 2020.
- **Bab 5 Implementasi Sistem**, merupakan bagian yang berisikan kode-kode untuk membangun Aplikasi Pendaftaran Peserta Didik Baru (PPDB) SMP, Kabupaten Bojonegoro 2020.
- **Bab 6 Kesimpulan dan Saran**, merupakan bagian yang berisikan tentang kesimpulan dan saran yang didapatkan dari hasil pengerjaan selama kerja praktik.

BAB 2

DESKRIPSI UMUM APLIKASI

2.1. Deskripsi Umum Sistem

Aplikasi Penerimaan Peserta Didik Baru (PPDB) SMP, Kabupaten Bojonegoro 2020 merupakan sebuah *website* sistem informasi yang digunakan untuk *monitoring* dan manajemen data yang diterima dari *website* pendaftaran dan pendataan calon siswa jenjang SMP di Kabupaten Bojonegoro, sehingga mempermudah administrasi dan pemantauan program pendidikan terlebih selama masa pandemi *Covid-19* ini.

Aplikasi Manajemen Penerimaan Peserta Didik Baru (PPDB) SMP, Kabupaten Bojonegoro 2020 digunakan oleh dua jenis pengguna, yaitu pihak sekolah SMP Negeri yang membuka pendaftaran dan penerimaan siswa didik baru tahun ajaran 2020/2021 dan pihak dinas pendidikan setempat yang digunakan untuk membantu dan manajemen program PPDB daring ini. Untuk memudahkan akses dan manajemen, pengembang juga memiliki akses terhadap sistem tersebut dengan akun administrator yang memiliki akses ke semua fitur yang ada di aplikasi ini. Tiap pengguna sistem Aplikasi

Manajemen Penerimaan Peserta Didik Baru (PPDB) SMP, Kabupaten Bojonegoro 2020 memiliki akses yang dapat diatur oleh akun administrator, yaitu dari pihak pengembang sendiri yang dapat mengubah hak akses semua pengguna.

Terdapat enam jenis jalur yang dibuka oleh Dinas Pendidikan Kabupaten Bojonegoro untuk tahun ajar 2020/2021, yaitu:

- **Jalur Afirmasi**, adalah jalur pendaftaran yang dibuka untuk calon siswa SMP yang memiliki kebutuhan finansial khusus dengan menyerahkan bukti berkas seperti Kartu Indonesia Pintar (KIP), Kartu Keluarga Sejahtera (KKS), Kartu Indonesia Sehat (KIS), Kartu Bantuan Pangan Non Tunai (KBPNT), atau Kartu Petani Mandiri (KPM).
- **Jalur Prestasi**, adalah jalur pendaftaran yang dibuka untuk calon siswa SMP yang memiliki bukti prestasi dibidang akademik atau non-akademik yang telah ditentukan oleh Dinas Pendidikan Kabupaten Bojonegoro dan pihak SMP Negeri. Bukti prestasi berupa hasil lomba seperti sertifikat atau penghargaan

dengan tingkat Internasional, Nasional, Provinsi, dan/atau Kabupaten/Kota yang diterbitkan paling singkat 6 (enam) bulan dan paling lama 3 (tiga) tahun sejak tanggal pendaftaran PPDB. Jalur Prestasi hanya diperuntukkan bagi calon peserta didik dari luar zona

- **Jalur Zonasi**, adalah jalur pendaftaran yang diperuntukkan bagi calon peserta didik dengan mempertimbangkan prinsip mendekatkan domisili (jarak) peserta didik dengan sekolah tujuan yang dibuktikan dengan Kartu Keluarga atau Surat Keterangan Domisili. Calon peserta didik jenjang SMP diberi kesempatan untuk mendaftar sesuai dengan zona tempat tinggalnya/domisili dan/atau luar Zona, pada Zona yang berbatasan. Data zonasi berdasarkan batas wilayah administrasi yang dituangkan dalam bentuk pengelompokan daerah (zona 1, zona 2, zona 3, zona 4, zona 5 dan zona 6) mulai dengan batas desa/kelurahan dan kecamatan.

- **Jalur Pindah Tugas Orang Tua**, diperuntukkan bagi calon peserta didik yang berdomisili di luar zona dibuktikan dengan surat penugasan dari instansi/lembaga pemerintah terhitung mulai tanggal 1 Juli 2019 sampai dengan pelaksanaan PPDB tahun 2020. Dalam hal kuota jalur perpindahan tugas orang tua belum terpenuhi, maka sisa kuota akan dimasukkan dalam jalur zonasi.
- **Jalur Siswa Tahun Lalu**, adalah jalur untuk Siswa Lulusan SD Swasta dan MI di Kabupaten Bojonegoro yang pihak Sekolahnya TIDAK MENGIRIMKAN Daftar Nominasi Tetap pada tanggal 14 Mei 2020 hingga 29 Mei 2020 ke Dinas Pendidikan Kabupaten Bojonegoro.
- **Jalur Siswa Luar Kota**, adalah jalur bagi siswa yang berasal dari luar Kabupaten Bojonegoro, Jawa Timur namun sudah berdomisili paling singkat 1 (satu) tahun sejak tanggal pendaftaran PPDB dengan menyerahkan bukti Kartu Keluarga (KK).

Setiap jalur memiliki tanggal pembukaan pendaftaran, penutupan, kuota, dan persyaratan tersendiri sesuai aturan yang telah disepakati dan ditetapkan bersama sekolah yang membuka pendaftaran PPDB.

2.2. Karakteristik Pengguna

Karakteristik pengguna dirinci dalam tabel sebagai berikut:

Tabel 2-1 Karakteristik pengguna aplikasi

Kategori Pengguna	Tugas	Hak Akses ke Aplikasi
Calon peserta didik	- Mengisi data pendaftaran	<ul style="list-style-type: none"> - Calon peserta didik dapat mengisi dan men-<i>submit</i> data pendaftaran. - Calon peserta didik dapat melihat status pendaftaran - Calon peserta didik dapat melihat hasil seleksi pendaftaran

Sekolah	<ul style="list-style-type: none"> - Melakukan manajemen dan <i>monitoring</i> sekolahnya 	<ul style="list-style-type: none"> - Sekolah dapat melihat daftar calon didik baru - Sekolah dapat menerima peserta didik baru - Sekolah dapat menolak peserta didik baru - Sekolah dapat mengirim notifikasi yang akan ditampilkan pada halaman pendaftaran dan pendataan calon peserta didik baru - Sekolah dapat mengunduh data peserta didik baru - Sekolah dapat melihat statistik jalur pendaftaran
Dinas	<ul style="list-style-type: none"> - Melakukan manajemen dan <i>monitoring</i> 	<ul style="list-style-type: none"> - Dinas dapat melihat data-data peserta - Dinas dapat menerima peserta didik baru

	<p>tiap sekolah</p>	<ul style="list-style-type: none"> - Dinas dapat menolak peserta didik baru - Dinas dapat mengunduh data peserta didik baru - Dinas dapat mengunduh data sekolah yang terdaftar akunnya - Dinas dapat mengubah akun sekolah - Dinas dapat melihat statistik tiap sekolah dan total pendaftaran
--	---------------------	---

2.3. Lingkungan Operasi

Lingkungan operasi untuk menjalankan Aplikasi Penerimaan Peserta Didik Baru (PPDB) SMP, Kabupaten Bojonegoro 2020 ini adalah sebagai berikut :

1. Aplikasi Penerimaan Peserta Didik Baru (PPDB) SMP, Kabupaten Bojonegoro 2020 berbasis *website* sehingga dapat diakses pada perangkat genggam atau laptop yang terhubung dengan internet.
2. Laptop atau komputer atau *smartphone* memiliki aplikasi *browser* terbaru.
3. *Smartphone* memiliki OS *Android* atau iOS sedangkan untuk laptop atau komputer menggunakan OS Windows, MacOS, atau Linux.

2.4. Batasan Perancangan dan Implementasi Sistem

Pengembangan aplikasi ini memiliki keterbatasan, yaitu sebagai berikut:

1. Aplikasi dibuat dengan menggunakan bahasa PHP, HTML, Javascript, CSS dan library seperti JQuery, Bootstrap, dan LeafletJS sebagai map. Database yang digunakan merupakan *mySQL*.
2. Hak akses terhadap aplikasi ini hanya terbatas untuk calon peserta didik, pihak sekolah, pihak dinas, dan tim pengembang saja.

BAB 3

PROSES BISNIS APLIKASI

Dalam pengembangan Aplikasi Penerimaan Peserta Didik Baru (PPDB) SMP, Kabupaten Bojonegoro 2020, memiliki alur bisnis yang dibagi menjadi tiga jenis hak akses yang berbeda, yaitu hak akses calon peserta didik, hak akses pihak sekolah, dan hak akses pihak dinas pendidikan. Pada aplikasi ini juga disediakan satu hak akses spesial untuk tim pengembang. Alur bisnisnya dimulai dari pihak dinas yang memberi data-data sekolah dan calon peserta didik kepada tim pengembang, kemudian tim pengembang memasukkan data ke dalam *database* sehingga pihak sekolah yang terdaftar memiliki akun untuk mengakses *website*, serta sebagai acuan untuk calon peserta didik dalam memilih sekolah. Secara umum, terdapat empat proses bisnis yang berjalan pada aplikasi ini, yaitu:

3.1 Proses 1

Pihak dinas memberi data ke tim pengembang berupa daftar sekolah dan daftar calon peserta didik untuk dilakukan proses data, seperti penghitungan jarak. Kemudian, tim

pengembang membuat akun untuk pihak calon peserta didik, pihak sekolah, dan pihak dinas. Jika ingin melakukan perubahan data, akun administrator yang dimiliki oleh tim pengembang dapat digunakan untuk mengatur dan mengubah data-data yang dimiliki oleh semua akun pada sistem. Pihak dari tim pengembang juga melakukan *monitoring* terhadap akun-akun beserta proses yang dilakukan seperti melakukan persetujuan-penolakan pendaftaran peserta didik baru. Data tersebut disimpan di dalam *log database* sehingga dapat di *monitoring* oleh tim pengembang dan pihak dinas.

3.2 Proses 2

Pihak calon peserta didik masuk ke sistem menggunakan akun yang telah disediakan oleh tim pengembang, atau menggunakan formulir khusus untuk membuat akun baru saat proses pendaftaran dimulai jika calon peserta didik belum terdaftar atau belum memiliki akun. Kemudian, calon peserta didik mengisi data yang dibutuhkan ke dalam formulir pendaftaran pada *website* pendaftaran dan pendataan calon peserta didik baru. Calon peserta didik dapat memantau secara berkala status pendaftarannya melalui *website*.

3.3 Proses 3

Pihak sekolah dapat melihat semua transaksi data yang didapatkan dari *website* pendaftaran dan pendataan calon peserta didik baru. Jika terdapat galat (*error*) ataupun ingin mengubah data pendaftaran, terdapat *log database* sehingga pihak administrator mengetahuinya. Jika permintaan untuk mengubah data pendaftaran calon peserta didik baru telah sesuai prosedur dan ketentuan dari pihak dinas, pihak dinas akan melaporkan ke perwakilan tim pengembang dan tim pengembang melakukan perubahan *database* sesuai permintaan.

3.4 Proses 4

Akun administrator memiliki hak akses penuh namun memiliki batasan untuk melakukan transaksi datanya, sesuai ketentuan dari pihak dinas. Akun administrator melakukan monitoring yang dilakukan meliputi *log update*, *log delete*, *log access server*, dan monitoring lainnya yang diperlukan sebagai bahan evaluasi ke depannya.

BAB 4

ANALISIS DAN DESAIN MODEL

4.1 Data Model Aplikasi PPDB SMP Kabupaten Bojonegoro 2020

Aplikasi PPDB SMP Kabupaten Bojonegoro 2020 memiliki tabel-tabel seperti pada Gambar 4.1-1.

Tabel-tabel pada Gambar 4.1-1 dijelaskan sebagai berikut:

1. Tabel sekolah dan pagu

Tabel ini berisi data-data sekolah, meliputi identitas, nama sekolah, zona, dan koordinat (*latitude* dan *longitude*). Kolom-kolom zona dan koordinat diperlukan untuk pertimbangan pengurutan calon peserta didik saat proses penerimaan. Setiap sekolah memiliki banyak jenis pagu yang disimpan di tabel pagu.

2. Tabel peserta

Tabel ini berisi data-data yang diperlukan untuk mencatat peserta, termasuk identitas, nama, NISN, alamat rumah, hingga koordinat tempat tinggal. Lokasi diperjelas dengan referensi ke tabel kelurahan, untuk pertimbangan zonasi. Pengurutan penerimaan berdasarkan koordinat yang telah dicantumkan

3. Tabel kelurahan, kecamatan, dan kelurahan zona

Tabel-tabel ini memuat data kelurahan dan kecamatan, serta untuk mencatat zona setiap kelurahan

4. Tabel pendaftaran, pilihan, verifikasi, pendaftaran prestasi, dan prestasi

Tabel ini berisi data pendaftaran calon peserta didik. Tabel ini memiliki banyak entri di tabel pilihan untuk menandakan sekolah mana saja yang dipilih. Tabel ini juga memiliki jenis pendaftaran jalur prestasi. Pendaftaran jalur prestasi mencantumkan prestasi yang diraih dalam tabel prestasi. Setiap entri pendaftaran memiliki satu baris yang bersesuaian di tabel verifikasi. Verifikasi dilakukan oleh petugas.

5. Tabel output rekomendasi sekolah

Tabel ini menghitung jarak setiap calon peserta didik ke setiap sekolah, untuk mempercepat proses penampilan jarak saat proses pendaftaran

6. Tabel *auth* peserta, *blacklist*

Tabel ini berisi data autentikasi seperti *username* dan *password*. Tabel *blacklist* berisi daftar-daftar NISN atau NIK yang tidak dapat mendaftar dalam aplikasi, karena alasan kesalahan data, atau alasan administrasi lain.

BAB 5

IMPLEMENTASI SISTEM

Pada bagian ini, penulis membahas beberapa potongan kode yang diimplementasikan ke dalam aplikasi PPDB SMP Kabupaten Bojonegoro 2020. Pembahasan akan dibagi menjadi dua bagian, yaitu bagian model bisnis dan bagian antarmuka pengguna. Sesuai penjelasan sebelumnya, aplikasi yang dikembangkan terdiri dari dua sistem yaitu aplikasi Pendaftaran dan Manajemen PPDB Kabupaten Bojonegoro 2020. Dua sistem tersebut menggunakan dua *framework* yang berbeda. Pada Pendaftaran digunakan *framework Laravel* dan pada Manajemen digunakan *framework CodeIgniter*.

5.1. Implementasi Antarmuka

5.1.1. Aplikasi Manajemen

5.1.1.1. Dashboard

Tabel 5.1-1 Kode UI Dashboard Aplikasi Manajemen

```
1. @extends('base/layout-dashboard')
2. @section('Judul', 'Dashboard Kantor PPDB Bojonego
   ro 2020')
3.
4. @section('content')
5.     <div class="breadcrumb-holder container-
      fluid">
6.         <ul class="breadcrumb">
7.             <li class="breadcrumb-
                item active"><a href="{ { base_url('/') }}">Dashbo
                ard</a></li>
8.         </ul>
9.     </div>
10.    <!-- Forms Section-->
11.    <section class="forms">
12.        <div class="container-fluid">
13.            @if($this->session-
                >flashdata('message'))
14.                @if($this->session-
                >flashdata('message')['type'] == 'error')
15.                    <div class="alert alert-
                        danger">
16.                        {{ implode('<br>', $this-
                >session->flashdata('message')['message']) }}
17.                    </div>
18.                @else
19.                    <div class="alert alert-
                        success">
20.                        {{ implode('<br>', $this-
                >session->flashdata('message')['message']) }}
21.                    </div>
22.                @endif
23.            @endif
24.
25.            {{ $informations }}
26.
```

```

27.         @if(in_array($this->session-
>user_login['role'], [ROLE_ADMIN, ROLE_DINAS, RO
LE_KADIN, ROLE_SEKOLAH]))
28.             <div class="row" id="loader-
main">
29.                 </div>
30.             <div class="row justify-content-
center" id="container-main">
31.                 <div class="col-6">
32.                     <canvas id="mainChart" wi
dth="200" height="200"></canvas>
33.                 </div>
34.             </div>
35.
36.             <div class="row" id="loader-
line">
37.                 </div>
38.             <div class="row justify-content-
center my-3" id="container-line">
39.                 <div class="col-6" id="block-
container-line">
40.                     <canvas id="totalPendafta
ranPerDay" width="200" height="200"></canvas>
41.                 </div>
42.             </div>
43.
44.             <div class="row mt-5">
45.                 <div class="container text-
center">
46.                     <h3>Jumlah Diterima dan D
itolak per Jalur</h3>
47.                 </div>
48.             </div>
49.             <div class="row">
50.                 <div class="form-group">
51.                     <label for="kategori">Kat
egori</label>
52.                     <select class="custom-
select" id="kategori">
53.                         <option selected valu
e="pendataan">Pendataan</option>

```

```

54.         <option value="pendaf
    taran">Pendaftaran</option>
55.         </select>
56.     </div>
57. </div>
58. <div class="row" id="loader-
    kategori">
59.     </div>
60. <div class="row" id="container-
    kategori">
61.
62.     </div>
63.
64.
65.         @endif
66.     </div>
67. </section>
68. @endsection
69.
70. @section('moreJs')
71.     <script src="https://cdn.jsdelivr.net/npm/chart.js@2.9.3/dist/Chart.min.js"></script>
72.     <script>
73.         const dict = {
74.             afirmasi: "Afirmasi",
75.             pindah_tugas: "Pindah Tugas",
76.             prestasi: "Prestasi",
77.             tahun_lalu: "Tahun Lalu",
78.             luar_kota: "Luar Kota",
79.             sd_mi: "Swasta / MI",
80.             zonasi: "Zonasi",
81.             tahap3: "Pemenuhan (Tahap 3)"
82.         }
83.
84.         const mainChart = (datas = [], labels = [
85.     ] => {
86.         let ctx = $('#mainChart');
87.         let chart = new Chart(ctx, {
88.             type: 'doughnut',
89.             data: {
90.                 labels: labels,

```



```

122.         data: {
123.             labels: labels,
124.             datasets: [{
125.                 label: 'Jumlah Sis
126.                 wa',
127.                 data: datas,
128.                 backgroundColor: [
129.                     'rgba(54, 162,
130.                     235, 1)',
131.                     'rgba(255, 99,
132.                     132, 1)',
133.                 ],
134.                 borderWidth: 1,
135.                 barPercentage: 0.8
136.             }],
137.             options: {
138.                 scales: {
139.                     yAxes: [{
140.                         ticks: {
141.                             beginAtZero: true
142.                         }
143.                     }],
144.                 },
145.                 title: {
146.                     display: true,
147.                     text: title,
148.                     fontSize: 14
149.                 },
150.                 legend: {
151.                     display: false
152.                 }
153.             }
154.         });
155.     }
156.     const totalPendaftaranPerDay = (ctx, datasets = [], labels = [], title = "") => {
157.         new Chart(ctx, {

```

```

157.         type: 'line',
158.         data: {
159.             labels,
160.             datasets,
161.         },
162.         options: {
163.             title: {
164.                 display: title !==
165.                 "",
166.                 text: title,
167.                 fontSize: 14
168.             },
169.             legend: {
170.                 display: true,
171.                 position: 'right'
172.             }
173.         });
174.     }
175.
176.     const init = async (kategori = 'pe
177.     ndataan') => {
178.         await Promise.all([
179.             (async () => getIndex())()
180.             ,
181.             (async (dataKategori) => g
182.             etKategori(kategori, dataKategori))(),
183.             (async () => getTotalPenda
184.             ftaranPerDay())(),
185.         ]);
186.     };
187.
188.     const getIndex = () => {
189.         $('#loader-
190.         main').append(`<div id="load-main" class="mx-
191.         auto">
192.             <div class="sp
193.             inner-grow text-info" role="status">
194.                 <span class="sr-only">Loading...</span>

```

```

189.                                     </div>
190.                                     </div>`);
191.     $.get('{{base_url('Visualisasi
Rest/index')}}', (data, status) => {
192.         $('#loader-
main').remove();
193.         const datasets = data.data
;
194.         let labels = [];
195.
196.         let tmpLabel = Object.keys
(datasets);
197.         tmpLabel.forEach(label =>
{
198.             labels.push(dict[label
])
199.         })
200.         let datas = Object.values(
datasets);
201.
202.         mainChart(datas, labels);
203.     });
204. }
205.
206.     const getTotalPendaftaranPerDay =
() => {
207.         const colours = [
208.             'rgba(255, 99, 132, 1)',
209.             'rgba(54, 162, 235, 1)',
210.             'rgba(255, 206, 86, 1)',
211.             'rgba(75, 192, 192, 1)',
212.             'rgba(153, 102, 255, 1)',
213.         ];
214.         $('#loader-
line').append(`<div id="load-total-
pendaftaran" class="mx-auto">
215.             <div class="sp
inner-grow text-info" role="status">
216.                 <span class="sr-only">Loading...</span>

```

```

217.                                     </div>
218.                                     </div>`);
219.         $.get('{{base_url('Visualisasi
Rest/totalPendaftaranPerDay')}}', {
220.             idSekolah: "<?php if (isse
t($this->session-
>user_login['sekolah_id'])) echo $this->session-
>user_login['sekolah_id'] ?>"
221.             }, (data, status) => {
222.                 $('#loader-
line').remove();
223.                 let titles = (Object.keys(
data.data)).map(label => dict[label]);
224.                 let labels = (Object.value
s(data.data)[0]).map(data => data.tgl_pendaftaran
);
225.                 let datasets = (Object.key
s(data.data)).map((jalur, idx) => {
226.                     return {
227.                         label: titles[idx]
228.                         ,
229.                         data: (Object.valu
es(data.data[jalur])).map(d => d.total),
230.                         fill: true,
231.                         showLine: true,
background-color: c
olours[idx]
232.                     }
233.                 })
234.                 totalPendaftaranPerDay$('
#totalPendaftaranPerDay'), datasets, labels, 'Tot
al Pendaftaran Per Hari (Data Sesuai Akun)');
235.                 });
236.             }
237.         }
238.
239.         const getKategori = (kategori = 'p
endataan') => {
240.             $('#loader-
kategori').append(`<div id="load-
pendataan" class="mx-auto">

```

```

241.                                     <div class="sp
      inner-grow text-info" role="status">
242.
      <span class="sr-only">Loading...</span>
243.                                     </div>
244.                                     </div>`);
245.     $.get('{{base_url('Visualisasi
      Rest/kategori')}}', {
246.         kategori
247.     }, function (data) {
248.         if (data.success) {
249.             $('#load-
      pendataan').remove();
250.             let labels = [];
251.             let tmpLabel = Object.
      keys(data.data);
252.             tmpLabel.forEach(label
      => {
253.                 labels.push(dict[l
      abel])
254.             });
255.
256.             labels.forEach((label,
      idx) => {
257.                 let container = $(
      `#${tmpLabel[idx]}`);
258.                 let label_chart =
      Object.keys(data.data[tmpLabel[idx]]);
259.                 let data_chart = O
      bject.values(data.data[tmpLabel[idx]]);
260.                 jalurChart(contain
      er, data_chart, label_chart, label);
261.             });
262.
263.         }
264.     });
265. }
266.
267. $(document).ready(function () {
268.     let kategori = 'pendataan';
269.     const pendataanHtml = `

```

```

270.         <div class="col-6 col-md-
271.             4">
272.             <canvas id="tahun_lalu
273.                 " width="200" height="200"></canvas>
274.             </div>
275.         <div class="col-6 col-md-
276.             4">
277.             <canvas id="luar_kota"
278.                 width="200" height="200"></canvas>
279.             </div>
280.         <div class="col-6 col-md-
281.             4">
282.             <canvas id="sd_mi" wid
283.                 th="200" height="200"></canvas>
284.             </div>
285.         `;
286.         `
287.         const pendaftaranHtml = `
288.         <div class="col-6 col-md-4">
289.             <canvas id="afirmasi"
290.                 width="200" height="200"></canvas>
291.             </div>
292.             <div class="col-6 col-md-
293.                 4">
294.                 <canvas id="pindah_tug
295.                     as" width="200" height="200"></canvas>
296.                 </div>
297.                 <div class="col-6 col-md-
298.                     4">
299.                     <canvas id="prestasi"
300.                         width="200" height="200"></canvas>
301.                     </div>
302.                     <div class="col-6 col-md-
303.                         4">
304.                         <canvas id="zonasi" wi
305.                             dth="200" height="200"></canvas>
306.                         </div>
307.                         <div class="col-6 col-md-
308.                             4">
309.                             <canvas id="tahap3" wi
310.                                 dth="200" height="200"></canvas>
311.                             </div>

```

```

297.         `;
298.
299.         $('#container-
    kategori').append(pendataanHtml);
300.
301.         init(kategori);
302.
303.         $('#kategori').change(function
    () {
304.             kategori = $(this).val();
305.             const container = $('#cont
    ainer-kategori');
306.             container.children().remov
    e();
307.
308.             if (kategori === 'pendataa
    n') {
309.                 container.append(penda
    taanHtml);
310.             } else if (kategori === 'p
    endaftaran') {
311.                 container.append(penda
    ftaranHtml);
312.             }
313.
314.             getKategori(kategori);
315.         });
316.
317.     })
318.
319.     </script>
320.
321.     @endsection

```

5.1.1.2. Daftar Ulang

Tabel 5.1-2 Kode UI Daftar Ulang

```

1. @extends('base/layout-dashboard')
2.

```

```

3. @section('Judul', 'Pemantauan Daftar Ulang Jalur
   '. ucfirst($data['jalur']))
4.
5. @section('moreCss')
6.
7. @endsection
8.
9. @section('content')
10. <div class="breadcrumb-holder container-
    fluid">
11.     <ul class="breadcrumb">
12.         <li class="breadcrumb-
            item"><a href="{{ base_url('/') }}">Dashboard</a>
            </li>
13.         <li class="breadcrumb-
            item active">Pemantauan Daftar Ulang {{ $jalur }}</
            li>
14.     </ul>
15. </div>
16.
17. <section>
18.     <div class="container-fluid">
19.         <div class="row">
20.             <div class="col">
21.                 @if($this->session-
                >flashdata('message'))
22.                     @if($this->session-
                >flashdata('message')['type'] == 'error')
23.                         <div class="alert ale
                rt-danger">
24.                             {{ implode('\n',
                $this->session-
                >flashdata('message')['message']) }}
25.                         </div>
26.                     @else
27.                         <div class="alert ale
                rt-success">
28.                             {{ implode('\n',
                $this->session-
                >flashdata('message')['message']) }}
29.                         </div>
30.                     @endif

```



```

31.         @endif
32.     </div>
33. </div>
34.
35.     <div class="row">
36.         <div class="col">
37.             <section class="dashboard-
counts p-0">
38.                 <div class="row bg-
white has-shadow">
39.
40.                     @if(count($belum)>0 &
& false)
41.                         <div class="col-
6 col-md-4">
42.                             <div class="i
tem d-flex align-items-center">
43.                                 <div clas
s="icon bg-warning text-white"><i
44.                                     class="fas fa-exclamation-triangle"></i>
45.                                 </div>
46.                                 <div clas
s="title"><span>Belum Verifikasi</span>
47.                                 <div
class="number"><strong>{{ sizeof($belum) }}</stro
ng></div>
48.                                 </div>
49.                             </div>
50.                         </div>
51.                     @endif
52.
53.                         <div class="col-
6 col-md-4">
54.                             <div class="item
d-flex align-items-center">
55.                                 <div class="i
con bg-danger text-white"><i class="fas fa-
times"></i></div>
56.                                 <div class="t
itle"><span>Belum Daftar Ulang</span>

```

```

57.         <div class="number"><strong>{{ sizeof($terima) }}</strong>
</div>
58.     </div>
59. </div>
60. </div>
61.
62.     <div class="col-6 col-md-4">
63.         <div class="item d-flex align-items-center">
64.             <div class="icon bg-success text-white"><i class="fa fa-check-circle"></i></div>
65.             <div class="title text-success"><span>Sudah Daftar Ulang</span>
66.             <div class="number"><strong>{{ sizeof($daful_online) }}</strong></div>
67.         </div>
68.     </div>
69. </div>
70.
71.     @if(false)
72.     <div class="col-6 col-md-4">
73.         <div class="item d-flex align-items-center">
74.             <div class="icon bg-info text-white"><i class="fas fa-info"></i></div>
75.             <div class="title text-info"><span>Sudah Daftar Ulang</span>
76.             <div class="number"><strong>{{ sizeof($terima) + sizeof($tolak) }}</strong>
77.         </div>
78.     </div>
79. </div>

```

```

80.                                     </div>
81.                                     @endif
82.
83.                                     <!-- Total -->
84.                                     <div class="col-
6 col-md-4">
85.                                     <div class="item
d-flex align-items-center">
86.                                     <div class="i
con bg-info text-white"><i class="fa fa-
users"></i></div>
87.                                     <div class="t
itle text-info">
88.                                     <span>Tot
al Siswa</span>
89.                                     <div clas
s="number">
90.                                     <stro
ng>{{ sizeof($daftar_online) + sizeof($terima) }}<
/strong>
91.                                     </div>
92.                                     </div>
93.                                     </div>
94.                                     </div>
95.                                     </div>
96.                                     </section>
97.                                     </div>
98.                                     </div>
99.
100.                                    @if(count($belum) > 0 && false
)
101.                                    <div class="row my-3">
102.                                    <div class="col-lg-
12 col-md-12 col-xs-12">
103.                                    <div class="card">
104.                                    <div class="ca
rd-header bg-dark text-
white">Daftar Siswa Belum Daftar Ulang</div>
105.                                    <div class="ca
rd-body">

```

```

106.                                     <div class
      ="table-responsive">
107.                                     <table
      class="table table-daful" id="belum-daful">
108.                                     <t
      head>
109.                                     <t
      r>
110.                                     <th>NISN</th>
111.                                     <th>Nama</th>
112.                                     @if($this->router-
      >fetch_method() !== 'tahap3')
113.                                     <th>Jarak (m)</th>
114.                                     @endif
115.                                     @if($this->router-
      >fetch_method() === 'prestasi')
116.                                     <th>Poin</th>
117.                                     @endif
118.                                     <th>Aksi</th>
119.                                     </
      tr>
120.                                     </
      thead>
121.                                     <t
      body>
122.                                     @f
      oreach($belum as $siswa)
123.                                     @f
124.                                     <tr>
125.                                     <td>{{$siswa->nisn}}</td>

```

```

126.         <td>{{ $siswa->nama_peserta }}</td>
127.         @if($this->router-
128.             >fetch_method() !== 'tahap3')
129.             <td>{{ $siswa->jarak }}</td>
130.         @endif
131.         @if(isset($siswa->poin))
132.             <td>{{ $siswa->poin }}</td>
133.         @endif
134.         <td>
135.             <a href="{{ base_url($endpoint . '/' . $
136.                 siswa->nisn . '/' . $siswa->id)}} "
137.                 style="width: 90%;"
138.                 class="btn btn-sm btn-
139.                     primary">Daftar Ulang</a>
140.         </td>
141.     </tr>
142. @endforeach
143. </tbody>
144. </table>
145. @endforeach
146. @endif
147.
148.

```

```

149.         <div class="row my-3">
150.             <div class="col-lg-12 col-
md-12 col-xs-12">
151.                 <div class="card">
152.                     <div class="card-
header bg-dark text-
white">Pemantauan Daftar Ulang</div>
153.                     <div class="card-
body">
154.                         <div class="ta
ble-responsive">
155.                             <table cla
ss="table table-daful" id="sudah-daful">
156.                                 <thead
>
157.                                     <tr>
158.                                         <t
h>NISN</th>
159.                                         <t
h>Nama</th>
160.                                         <t
h>Nomor HP Orang Tua</th>
161.                                         @i
f($this->router->fetch_method() !== 'tahap3')
162.                                         <th>Jarak (m)</th>
163.                                         @e
ndif
164.                                         @i
f($this->router->fetch_method() === 'prestasi')
165.                                         <th>Poin</th>
166.                                         @e
ndif
167.                                         <t
h>Sudah Daftar Ulang Online</th>
168.                                         <t
h>Aksi</th>
169.                                     </tr>
170.                                 </thea
d>

```

```

171.
172.                                     <tbody
173.     >
174.                                     <!--
175.     - Belum Daful -->
176.                                     @forea
177.     ch($terima as $siswa)
178.                                     <t
179.     r>
180.     <td>{{ $siswa->nisn}}</td>
181.     <td>{{ $siswa->nama_peserta}}</td>
182.     @if(isset($siswa->no_hp))
183.         <td>{{ $siswa->no_hp}}</td>
184.     @else
185.         <td>-</td>
186.     @endif
187.     @if($this->router-
188.     >fetch_method() !== 'tahap3')
189.         <td>{{ $siswa->jarak}}</td>
190.     @endif
191.     @if(isset($siswa->poin))
192.         <td>{{ $siswa->poin}}</td>
193.     @endif
194.     <td>

```

```

192.         <button class="btn btn-sm btn-
warning disabled">Belum</button>
193.     </td>
194.
195.     @if(false)
196.         @if($siswa-
>jalur == JALUR_ZONASI || $siswa-
>jalur == JALUR_TAHAP3)
197.             <td>
198.                 <a href="{{base_url(sprintf('dashbo
ard/lihatBerkas/%s/%s/%s/%s', $siswa-
>nik, $siswa->nisn, $siswa->jalur, $siswa-
>id_pendaftaran))}}"
199.                     class="btn btn-sm btn-
info">Cek Berkas Siswa</a>
200.             </td>
201.             @else
202.             <td>
203.                 <a href="{{base_url(sprintf('dashbo
ard/lihatBerkas/%s/%s/%s', $siswa->nik, $siswa-
>nisn, $siswa->jalur))}}"
204.                     class="btn btn-sm btn-
info">Cek Berkas Siswa</a>
205.             </td>
206.             @endif
207.         @endif

```



```

208.     <td>
209.         <a href="{{base_url(sprintf('%s/%s/%s/%s',
210.             $endpoint,$siswa->nisn, $siswa-
211.             >id_pendaftaran, true))}}"
212.             class="btn btn-sm btn-info">
213.                 Cek Berkas Siswa
214.             </a>
215.         </td>
216.     </tr>
217. @endfo
218. reach
219. @forea
220. ch($dافل_online as $siswa)
221.     <tr>
222.         <td>{{ $siswa->nisn }}</td>
223.         <td>{{ $siswa->nama_peserta }}</td>
224.         @if(isset($siswa->no_hp))
225.             <td>{{ $siswa->no_hp }}</td>
226.         @else
227.             <td>-</td>
228.         @endif

```

```

228.     @if($this->router-
>fetch_method() != 'tahap3')
229.         <td>{{$siswa->jarak}}</td>
230.     @endif
231.
232.     @if(isset($siswa->poin))
233.         <td>{{$siswa->poin}}</td>
234.     @endif
235.     <td>
236.         <button class="btn btn-sm btn-
success disabled">Sudah
237.     </button>
238. </td>
239.
240.     @if(false)
241.         @if($siswa-
>jalur == JALUR_ZONASI || $siswa-
>jalur == JALUR_TAHAP3)
242.             <td>
243.                 <a href="{{base_url(sprintf('dashbo
ard/lihatBerkas/%s/%s/%s/%s', $siswa-
>nik, $siswa->nisn, $siswa->jalur, $siswa-
>id_pendaftaran))}}"
244.                     class="btn btn-sm btn-
info">Cek Berkas Siswa</a>
245.             </td>

```

```

246.         @else
247.             <td>
248.                 <a href="{{base_url(sprintf('dashbo
ard/lihatBerkas/%s/%s/%s', $siswa->nik, $siswa-
>nis, $siswa->jalur))}}"&
249.                     class="btn btn-sm btn-
info">Cek Berkas Siswa</a>
250.             </td>
251.         @endif
252.     @endif
253. <td>
254.     <a href="{{base_url(sprintf('%s/%s/%s/%s',
$endpoint,$siswa->nis, $siswa-
>id_pendaftaran, true))}}"&
255.         class="btn btn-sm btn-info">
256.         Cek Berkas Siswa
257.     </a>
258. </td>
259. </
tr>
260. @endfo
reach
261.
262. @if(fa
lse)
263. @forea
ch($tolak as $siswa)
264. <t
r>

```

```

265.     <td>{{ $siswa->nisn }}</td>
266.     <td>{{ $siswa->nama_peserta }}</td>
267.     @if(isset($siswa->no_hp))
268.         <td>{{ $siswa->no_hp }}</td>
269.     @else
270.         <td>-</td>
271.     @endif
272.     @if($this->router-
273.         >fetch_method() !== 'tahap3')
274.         <td>{{ $siswa->jarak }}</td>
275.     @endif
276.     @if(isset($siswa->poin))
277.         <td>{{ $siswa->poin }}</td>
278.     @endif
279.     <td>
280.         <button class="btn btn-sm btn-
281.             danger disabled">Ditolak</button>
282.     </td>
283.     @if(false)
284.         @if($siswa-
285.             >jalur == JALUR_ZONASI || $siswa-
286.             >jalur == JALUR_TAHAP3)

```

```

284.         <td>
285.             <a href="{{base_url(sprintf('dashbo
ard/lihatBerkas/%s/%s/%s/%s', $siswa-
>nik, $siswa->nisn, $siswa->jalur, $siswa-
>id_pendaftaran))}}"
286.                 class="btn btn-sm btn-
info">Cek Berkas Siswa</a>
287.         </td>
288.     @else
289.         <td>
290.             <a href="{{base_url(sprintf('dashbo
ard/lihatBerkas/%s/%s/%s', $siswa->nik, $siswa-
>nisn, $siswa->jalur))}}"
291.                 class="btn btn-sm btn-
info">Cek Berkas Siswa</a>
292.         </td>
293.     @endif
294. @endif
295. <td>
296.     <a href="{{base_url(sprintf('%s/%s/%s/%s',
$endpoint,$siswa->nisn, $siswa-
>id_pendaftaran, true))}}"
297.         class="btn btn-sm btn-info">
298.         Cek Berkas Siswa
299.     </a>

```

```

300.         </td>
301.                                     </
tr>
302.                                     @endfo
reach
303.                                     @endif
304.                                     </tbody>
y>
305.                                     </table>
306.                                     </div>
307.                                     </div>
308.                                     </div>
309.                                     </div>
310.                                     </div>
311.                                     </div>
312.                                     </section>
313.
314.
315.         @endsection
316.
317.         @section('moreJs')
318.             <script>
319.
320.                 $(document).ready(function () {
321.                     $('table-
daful').DataTable({
322.                         "language": {
323.                             "url": "https://cdn.datatables
es.net/plug-ins/1.10.21/i18n/Indonesian.json"
324.                         },
325.                         @if($this->router-
>fetch_method() === 'prestasi')
326.                             "order": [[4, 'desc'], [3,
'asc']]
327.                         @elseif($this->router-
>fetch_method() === 'tahap3')
328.                             "order": [[1, 'asc']]
329.                         @else
330.                             "order": [[3, 'asc']]
331.                         @endif

```

```

332.         });
333.
334.
335.         });
336.
337.     </script>
338. @endsection

```

5.1.1.3. Lihat Berkas Pendaftar (Admin)

Tabel 5.1-3 Kode UI Lihat Berkas untuk akun administrator

```

1. @extends('base/layout-dashboard')
2.
3. @section('Judul', 'Lihat Berkas Siswa')
4.
5. @section('moreCss')
6.     <style>
7.         li {
8.             list-style-type: none;
9.         }
10.     </style>
11. @endsection
12.
13. @section('content')
14.     <div class="breadcrumb-holder container-
15.         fluid">
16.         <ul class="breadcrumb">
17.             <li class="breadcrumb-
18.                 item"><a href="{{base_url('/')}}">Dashboard</a></
19.                 li>
20.             <li class="breadcrumb-
21.                 item active">Lihat Berkas Siswa</li>
22.         </ul>
23.     </div>
24.
25.     <section>
26.         <div class="container-fluid">
27.             <div class="card">

```

```

25.         <div class="card-header bg-
dark text-white">Berkas Siswa</div>
26.
27.         <div class="card-body">
28.             <label for="pilihan-
jalur">Kategori</label>
29.             <select class="browser-
default custom-select" id="pilihan-jalur">
30.                 <option selected value="n
on-zonasi">Non-zonasi</option>
31.                 <option value="zonasi">Zo
nasi / Tahap 3</option>
32.
33.             </select>
34.
35.             <section id="non-zonasi">
36.                 <div class="md-form mt-
0">
37.                     <input class="form-
control" id="nik" type="text" placeholder="NIK" a
ria-label="NIK">
38.                     <input class="form-
control" id="nisn" type="text" placeholder="NISN"
aria-label="NISN">
39.                 </div>
40.                 <label for="jalur">Jalur<
/label>
41.                 <select class="custom-
select" id="jalur">
42.                     </select>
43.                 </section>
44.
45.             <section id="zonasi" hidden>
46.                 <div class="md-form mt-
0">
47.                     <input class="form-
control" id="idPendaftaran" placeholder="ID Penda
ftaran"
48.                     aria-
label="ID Pendaftaran">
49.                 </div>

```



```

50.         </section>
51.
52.
53.         <button id="btnSearch" class=
"btn btn-primary">Search</button>
54.         </div>
55.     </div>
56.
57.     <div class="card">
58.         <div class="card-
body" id="files">
59.             <div class="col" id="file-
container">
60.                 </div>
61.             </div>
62.         </div>
63.
64.     </div>
65.
66. </section>
67.
68.
69.
70. @endsection
71.
72. @section('moreJs')
73.     <script>
74.         const data = [
75.             {
76.                 text: 'Tahun Lalu',
77.                 id: 'tahun-lalu',
78.                 element: HTMLOptionElement
79.             },
80.             {
81.                 text: 'Luar Kota',
82.                 id: 'luar-kota',
83.                 element: HTMLOptionElement
84.             },
85.             {
86.                 text: 'Swasta - MI',
87.                 id: 'swasta-mi',
88.                 element: HTMLOptionElement

```

```

89.         },
90.         {
91.             text: 'Prestasi',
92.             id: 'prestasi',
93.             element: HTMLOptionElement
94.         },
95.         {
96.             text: 'Afirmasi',
97.             id: 'afirmasi',
98.             element: HTMLOptionElement
99.         },
100.        {
101.            text: 'Pindah Tugas',
102.            id: 'pindah-tugas',
103.            element: HTMLOptionElement
104.        }
105.    ];
106.
107.    $('#jalur').select2({
108.        data: data
109.    });
110.
111.    const nonZonasi = (viewer) => {
112.        let jalur = $('#jalur').val();
113.
114.        let nik = $('#nik').val()
115.        let nisp = $('#nisp').val();
116.
117.        if (jalur !== '' && nik !== ''
118.            && nisp !== '') {
119.            $('#files').append("<div id=
120.            'load' class='\"spinner-grow text-
121.            center\" role='\"status\">\n" +
122.            " <span class='\"s
123.            r-only\">Loading...</span>\n" +
124.            "</div>");
125.
126.            $.post('{{base_url('FetchA
127.            pi/berkasSiswa')}}', {
128.                jalur: jalur,
129.                nik: nik,

```

```

124.             nisn: nisn
125.             }, (data, status) => {
126.                 if (status === 500 ||
data.data.length === 0) {
127.                     window.alert('Error
r 500');
128.                     $('#load').remove(
);
129.                 } else {
130.                     data.data.forEach(
d => {
131.                         if (d !== 'dat
a:') {
132.                             let img =
`<img src=${d} class="img-responsive" style="max-
width: 200px;" alt="file"/>`
133.                             $('#file-
container')
134.                             .a
ppend(img)
135.                             }
136.                             $('#load').rem
ove();
137.                             });
138.                             viewer = new Viewe
r(document.getElementById('file-container'));
139.                             viewer.show();
140.                             }
141.                             })
142.                             }
143.                             };
144.
145.                             const zonasi = (viewer) => {
146.                                 const idPendaftaran = $('#idPe
ndaftaran').val();
147.
148.                                 if (idPendaftaran !== '') {
149.                                     $('#files').append("<div i
d='load' class=\"spinner-grow text-
center\" role=\"status\">\n" +
150.                                     " <span class=\"s
r-only\">Loading...</span>\n" +

```

```

151.         "</div>");
152.
153.         $.post('{{base_url('FetchA
pi/berkasSiswaZonasi')}}', {
154.             idPendaftaran,
155.             jalur: "{{$jalur}}"
156.         }, (data, status) => {
157.             if (status === 500 ||
data.data.length === 0) {
158.                 window.alert('Error
r 500');
159.                 $('#load').remove(
);
160.             } else {
161.                 data.data.forEach(
d => {
162.                     if (d !== 'dat
a:') {
163.                         let img =
`<img src=${d} class="img-responsive" style="max-
width: 200px;" alt="file"/>`
164.                         $('#file-
container')
165.                             .a
ppend(img)
166.                             }
167.                         $('#load').rem
ove();
168.                     });
169.                     viewer = new Viewe
r(document.getElementById('file-container'));
170.                     viewer.show();
171.                 }
172.             });
173.         }
174.     };
175.
176.     $(document).ready(function () {
177.         let pilihan = 'non-zonasi';
178.         let viewer = new Viewer(docume
nt.getElementById('file-container'));
179.

```

```

180.         @if(isset($nik))
181.         $('#nik').attr('value', '{{ $nik
           k}}');
182.         @endif
183.
184.         @if(isset($nisn))
185.         $('#nisn').attr('value', '{{ $nisn
           isn}}');
186.         @endif
187.
188.         @if(isset($idPendaftaran))
189.         $('#idPendaftaran').attr('valu
           e', '{{ $idPendaftaran}}');
190.         $('#pilihan-
           jalur').val('zonasi');
191.         $('#section#non-
           zonasi').attr('hidden', true);
192.         $('#section#zonasi').attr('hidd
           en', false);
193.         @endif
194.
195.         @if(isset($jalur))
196.         $('#jalur').val('{{ $jalur}}').
           trigger('change');
197.         @endif
198.
199.         $('#pilihan-
           jalur').change(function () {
200.             pilihan = $(this).val();
201.
202.             if (pilihan === 'non-
           zonasi') {
203.                 $('#section#zonasi').at
           tr('hidden', true);
204.                 $('#section#non-
           zonasi').attr('hidden', false);
205.             } else if (pilihan === 'zo
           nasi') {
206.                 $('#section#non-
           zonasi').attr('hidden', true);
207.                 $('#section#zonasi').at
           tr('hidden', false);

```

```

208.         }
209.     });
210.
211.     $('#btnSearch').click(function
212.     () {
213.         pilihan = $('#pilihan-
214.         jalur').val();
215.         $('#file-
216.         container > img').remove();
217.         $('#load').remove();
218.         viewer.destroy();
219.         if (pilihan === 'non-
220.         zonasi') {
221.             nonZonasi(viewer);
222.         } else if (pilihan === 'zo
223.         nasi') {
224.             zonasi(viewer);
225.         }
226.     });
227. });
228.
229.     </script>
230. @endsection

```

5.2. Implementasi Model Bisnis

Implementasi model bisnis akan berisikan kelas-kelas yang digunakan beserta beberapa potongan query yang digunakan untuk baca-tulis database. Juga akan ditampilkan beberapa controller yang menjadi perantara dan pengatur antara antarmuka dengan database/framework secara keseluruhan.

5.2.1. Aplikasi Manajemen

5.2.1.1. Model Autentifikasi

Tabel 5.2-1 Kode Model Autentifikasi

```
1. <?php
2.
3. class Autentifikasi extends CI_Model
4. {
5.     private const table = "user_kantor";
6.
7.     public function get_role($table)
8.     {
9.         $this->read_db_cluster->select("*");
10.        $this->read_db_cluster-
11.        >from($table);
12.        return $this->read_db_cluster->get()-
13.        >result();
14.    }
15.
16.    public function insert_user($table, $data)
17.    {
18.        $this->write_db_cluster-
19.        >trans_start();
```

```

18.         $stats = $this->write_db_cluster-
>insert($table, $data);
19.         $this->write_db_cluster-
>trans_complete();
20.
21.         return $stats;
22.     }
23.
24.     public function updateLastLogin($username,
    $datetime)
25.     {
26.         $this->write_db_cluster-
>trans_start();
27.         $this->write_db_cluster-
>where('username', $username);
28.         $this->write_db_cluster-
>update(self::table, array('last_login' => $da
tetime));
29.         $this->write_db_cluster-
>trans_complete();
30.
31.         return $this->write_db_cluster-
>trans_status() === true;
32.     }
33.
34.     public function deactivateUserByUsername($
username)
35.     {
36.         $table = "user_kantor";
37.         $this->write_db_cluster-
>trans_start();
38.         $this->write_db_cluster-
>update($table, array('active' => 0));
39.         $this->write_db_cluster-
>where('username', $username);
40.         $this->write_db_cluster-
>trans_complete();
41.
42.         return $this->write_db_cluster-
>trans_status() === true;
43.     }
44.

```



```

45.     public function activateUserByUsername($us
        ername)
46.     {
47.         $this->write_db_cluster-
>trans_start();
48.         $this->write_db_cluster-
>where('username', $username);
49.         $this->write_db_cluster-
>update(self::table, array('active', 1));
50.         $this->write_db_cluster-
>trans_complete();
51.
52.         return $this->write_db_cluster-
>trans_status() === true;
53.     }
54.
55.     function check_login($table, $field1, $fie
        ld2)
56.     {
57.         $this->read_db_cluster-
>select($table . '.*, role_users.*,' . $table
        . '.id as id_user');
58.         $this->read_db_cluster-
>from($table);
59.         $this->read_db_cluster-
>where($field1);
60.         $this->read_db_cluster-
>where($field2);
61.         $this->read_db_cluster->limit(1);
62.         $this->read_db_cluster-
>join('role_users', $table . '.role_id = role_
        users.id');
63.         $query = $this->read_db_cluster-
>get();
64.         if ($query->num_rows() == 0) {
65.             return FALSE;
66.         } else {
67.             return $query->first_row();
68.         }
69.     }
70.

```

```

71.     function check_login_smp($table, $field1,
72.     $field2, $id_role)
73.     {
74.         $this->read_db_cluster-
75.         >select($table . '.*, kota_kabupaten.*, role_u
76.         sers.*, ' . $table . '.id as id_user');
77.         $this->read_db_cluster-
78.         >from($table);
79.         $this->read_db_cluster-
80.         >where($field1);
81.         $this->read_db_cluster-
82.         >where($field2);
83.         $this->read_db_cluster-
84.         >where('users2.id_role', $id_role);
85.         $this->read_db_cluster->limit(1);
86.         $this->read_db_cluster-
87.         >join('role_users', $table . '.id_role = role_
88.         users.id_role');
89.         $this->read_db_cluster-
90.         >join('kota_kabupaten', $table . '.kode_kota =
91.         kota_kabupaten.kode_kota_kabupaten', 'left');
92.
93.         $query = $this->read_db_cluster-
94.         >get();
95.         if ($query->num_rows() == 0) {
96.             return FALSE;
97.         } else {
98.             return $query->first_row();
99.         }
100.     }
101.
102.     public function findByUsername($table, $us
103.     ername)
104.     {
105.         $this->read_db_cluster-
106.         >select($table . '.*, kota_kabupaten.*, role_u
107.         sers.*, ' . $table . '.id as id_user');
108.         $this->read_db_cluster-
109.         >from($table);
110.         $this->read_db_cluster-
111.         >where('username', $username);
112.         $this->read_db_cluster->limit(1);

```

```

95.         $this->read_db_cluster-
>join('role_users', $table . '.id_role = role_
users.id_role');
96.         $this->read_db_cluster-
>join('kota_kabupaten', $table . '.kode_kota =
kota_kabupaten.kode_kota_kabupaten', 'left');

97.         $this->read_db_cluster->limit(1);
98.         $query = $this->read_db_cluster-
>get();
99.
100.                return $query->first_row();
101.
102.                }
103.        }

```

5.2.1.2. Model Pendaftaran

Tabel 5.2-2 Model Pendaftaran pada Aplikasi Manajemen

```

1. <?php
2.
3. class Pendaftaran extends CI_Model
4. {
5.     // Mendapatkan daftar pendaftaran prestasi
6.     public function get_prestasi($status, $sekola
hId = '')
7.     {
8.         $this->read_db_cluster-
>select('pendaftaran.id as pendaftaran_id, pendaf
taran.*, peserta.*');
9.         $this->read_db_cluster-
>from('pendaftaran');
10.        $this->read_db_cluster-
>where('pendaftaran.status', $status);
11.        $this->read_db_cluster-
>join("pendaftaran_prestasi", "pendaftaran_presta
si.id_pendaftaran = pendaftaran.id");

```

```

12.     $this->read_db_cluster-
    >join("peserta", "pendaftaran.id_peserta = pesert
    a.id");
13.
14.     // Hanya diverifikasi oleh sekolah pilihaha
    n pertama
15.     if (!empty($sekolahId)) {
16.         $this->read_db_cluster-
    >join('pilihan', 'pilihan.id_pendaftaran = pendaf
    taran.id');
17.         $this->read_db_cluster-
    >where('pilihan.id_sekolah', $sekolahId);
18.         $this->read_db_cluster-
    >where('pilihan.indeks_pilihan', 1); // Dapatkan
    pilihan 1 aja
19.     }
20.
21.     return $this->read_db_cluster->get()-
    >result();
22. }
23.
24. // Mendapatkan daftar sekolah yang telah dipi
    lih
25. public function get_daftar_pilihan($id_pendaft
    aran)
26. {
27.     $this->read_db_cluster-
    >select('sekolah.nama_sekolah, sekolah.id as id_s
    ekolah');
28.     $this->read_db_cluster-
    >from('pilihan');
29.     $this->read_db_cluster-
    >where('pilihan.id_pendaftaran', $id_pendaftaran)
    ;
30.     $this->read_db_cluster->limit(5);
31.     $this->read_db_cluster-
    >order_by('pilihan.indeks_pilihan');
32.     $this->read_db_cluster-
    >join("sekolah", "pilihan.id_sekolah = sekolah.id
    ");
33.

```

```

34.         return $this->read_db_cluster->get()-
>result();
35.     }
36.
37.     public function getDaftarPilihanPindahTugas($
nik)
38.     {
39.         $this->read_db_cluster-
>select('sekolah.nama_sekolah, sekolah.id as id_s
ekolah');
40.         $this->read_db_cluster-
>from('peserta_pindah_tugas');
41.         $this->read_db_cluster-
>join('sekolah', 'sekolah.id = peserta_pindah_tug
as.id_sekolah');
42.         $this->read_db_cluster-
>where('peserta_pindah_tugas.nik', $nik);
43.
44.         return $this->read_db_cluster->get()-
>result();
45.     }
46.
47.     public function getDaftarPilihanTahap3($nisp)
48.     {
49.         $this->read_db_cluster-
>select('sekolah.nama_sekolah, sekolah.id as id_s
ekolah');
50.         $this->read_db_cluster-
>from('peserta_tahap3');
51.         $this->read_db_cluster-
>join('sekolah', 'sekolah.id = peserta_tahap3.id_
sekolah');
52.         $this->read_db_cluster-
>where('peserta_tahap3.nisp', $nisp);
53.
54.         return $this->read_db_cluster->get()-
>result()[0];
55.     }
56.
57.     // END daftar sekolah pilihan
58.

```

```

59. // Mendapatkan daftar pendaftaran
60. public function get_pendaftaran($jalur, $status, $sekolahId = '')
61. {
62.     if ($jalur !== JALUR_PERPINDAHAN_TUGAS) {
63.         // Hanya diverifikasi oleh sekolah pilihan pertama
64.         if (!empty($sekolahId)) {
65.             $t1 = 'pendaftaran.id as pendaftaran_id, pendaftaran.*, sekolah.latitude as sekolah_lat, sekolah.longitude as sekolah_long';
66.             $t2 = 'peserta.*';
67.             $select = $t1 . ', ' . $t2;
68.
69.             $this->read_db_cluster->select($select);
70.             $this->read_db_cluster->from('pendaftaran');
71.             $this->read_db_cluster->where('pendaftaran.tipe_pendaftaran', $jalur);
72.             $this->read_db_cluster->where('pendaftaran.status', $status);
73.             $this->read_db_cluster->join("peserta", "pendaftaran.id_peserta = peserta.a.id");
74.
75.             $this->read_db_cluster->join('pilihan', 'pilihan.id_pendaftaran = pendaftaran.id');
76.             $this->read_db_cluster->join('sekolah', 'sekolah.id = pilihan.id_sekolah');
77.             $this->read_db_cluster->where('pilihan.id_sekolah', $sekolahId);
78.             $this->read_db_cluster->where('pilihan.indeks_pilihan', 1); // Dapatkan pilihan 1 aja
79.
80.         } else {
81.             $t1 = 'pendaftaran.id as pendaftaran_id, pendaftaran.*';

```

```

82.             $t2 = 'peserta.*';
83.             $select = $t1 . ', ' . $t2;
84.
85.             $this->read_db_cluster-
>select($select);
86.             $this->read_db_cluster-
>from('pendaftaran');
87.             $this->read_db_cluster-
>where('pendaftaran.tipe_pendaftaran', $jalur);
88.             $this->read_db_cluster-
>where('pendaftaran.status', $status);
89.             $this->read_db_cluster-
>join("peserta", "pendaftaran.id_peserta = pesert
a.id");
90.         }
91.     } else {
92.         $t1 = 'pendaftaran.id as pendaftaran_
id, pendaftaran.*';
93.         $t2 = 'peserta_pindah_tugas.*';
94.         $select = $t1 . ', ' . $t2;
95.
96.         $this->read_db_cluster-
>select($select);
97.         $this->read_db_cluster-
>from('pendaftaran');
98.         $this->read_db_cluster-
>where('pendaftaran.tipe_pendaftaran', $jalur);
99.         $this->read_db_cluster-
>where('pendaftaran.status', $status);
100.        $this->read_db_cluster-
>join("peserta", "pendaftaran.id_peserta = pesert
a.id");
101.        $this->read_db_cluster-
>join('peserta_pindah_tugas', 'peserta_pindah_tug
as.nik = peserta.nik');
102.
103.         if (!empty($sekolahId)) {
104.             $this->read_db_cluster-
>where('peserta_pindah_tugas.id_sekolah', $sekola
hId);
105.         }

```

```

106.         }
107.
108.         return $this->read_db_cluster-
>get()->result();
109.     }
110.
111.     public function getPesertaPindahTugas(
    $status, $sekolahId = '')
112.     {
113.         $this->read_db_cluster-
>select('peserta_pindah_tugas.id as pendaftaran_i
d, peserta_pindah_tugas.*');
114.         $this->read_db_cluster-
>from('peserta_pindah_tugas');
115.         $this->read_db_cluster-
>where('status', $status);
116.
117.         if (!empty($sekolahId)) {
118.             $this->read_db_cluster-
>where('id_sekolah', $sekolahId);
119.         }
120.
121.         return $this->read_db_cluster-
>get()->result();
122.     }
123.
124.     // END daftar pendaftaran
125.
126.     // Mendapatkan biodata siswa
127.     public function get_siswa($nik, $jalur
, $idPendaftaran)
128.     {
129.         if ($jalur == JALUR_PRESTASI) {
130.             $this->read_db_cluster-
>select('pendaftaran.id as pendaftaran_id, pesert
a.*,
131.         pendaftaran_prestasi.id as pen
daftaran_prestasi_id, pendaftaran_prestasi.poin_t
otal, kelurahan.nama_kelurahan, kecamatan.nama_ke
camatan');
132.             $this->read_db_cluster-
>from('peserta');

```



```

133.             $this->read_db_cluster-
>join("pendaftaran", "pendaftaran.id_peserta = pe
serta.id");
134.             $this->read_db_cluster-
>join("pendaftaran_prestasi", "pendaftaran_presta
si.id_pendaftaran = pendaftaran.id");
135.             $this->read_db_cluster-
>where('peserta.nik', $nik);
136.             //$this->read_db_cluster-
>limit(1);
137.             $this->read_db_cluster-
>where('pendaftaran.id', $idPendaftaran);
138.             } // Afirmasi
139.             else {
140.                 $this->read_db_cluster-
>select('pendaftaran.id as pendaftaran_id, pesert
a.*, kelurahan.nama_kelurahan, kecamatan.nama_kec
amatan');
141.                 $this->read_db_cluster-
>from('pendaftaran');
142.                 $this->read_db_cluster-
>join("peserta", "peserta.id = pendaftaran.id_pes
erta");
143.                 $this->read_db_cluster-
>where('pendaftaran.id', $idPendaftaran);
144.                 //$this->read_db_cluster-
>limit(1);
145.                 $this->read_db_cluster-
>where('peserta.nik', $nik);
146.             }
147.
148.             $this->read_db_cluster-
>join('kelurahan', "peserta.id_kelurahan = kelura
han.id");
149.             $this->read_db_cluster-
>join('kecamatan', "kelurahan.id_kecamatan = keca
matan.id");
150.
151.             // Kalau double
152.             $this->read_db_cluster-
>order_by('pendaftaran.created_at', 'DESC');
153.

```

```

154.         return $this->read_db_cluster-
>get()->result();
155.     }
156.
157.     public function getSiswaPindahTugas($n
ik, $id)
158.     {
159.         $this->read_db_cluster-
>select('peserta_pindah_tugas.id as id_peserta_pi
ndah_tugas, peserta_pindah_tugas.*');
160.         $this->read_db_cluster-
>from('peserta_pindah_tugas');
161.         $this->read_db_cluster-
>where('peserta_pindah_tugas.nik', $nik);
162.         $this->read_db_cluster-
>where('peserta_pindah_tugas.id', $id);
163.
164.         // Kalau double
165.         $this->read_db_cluster-
>order_by('peserta_pindah_tugas.created_at', 'DES
C');
166.
167.         return $this->read_db_cluster-
>get()->result();
168.     }
169.
170.     public function getSiswaPindahTugasByI
d($id)
171.     {
172.         $this->read_db_cluster-
>select('peserta_pindah_tugas.*');
173.         $this->read_db_cluster-
>from('peserta_pindah_tugas');
174.         $this->read_db_cluster-
>where('peserta_pindah_tugas.id', $id);
175.
176.         // Kalau double
177.         $this->read_db_cluster-
>order_by('peserta_pindah_tugas.created_at', 'DES
C');
178.

```

```

179.         return $this->read_db_cluster-
>get()->result();
180.     }
181.
182.         // END biodata siswa
183.
184.     public function get_list_prestasi($id_
pendaftaran_prestasi)
185.     {
186.         $this->read_db_cluster-
>select('*');
187.         $this->read_db_cluster-
>from('prestasi');
188.         $this->read_db_cluster-
>where('id_pendaftaran_prestasi', $id_pendaftaran_
prestasi);
189.
190.         return $this->read_db_cluster-
>get()->result();
191.     }
192.
193.     public function getByNikNisn($nik, $ni
sn, $select = null)
194.     {
195.         if (!empty($select)) {
196.             $this->read_db_cluster-
>select($select);
197.             if ($select == 'peserta_pindah
_tugas.attachment') {
198.                 $this->read_db_cluster-
>from('peserta_pindah_tugas');
199.                 $this->read_db_cluster-
>where('peserta_pindah_tugas.nik', $nik);
200.                 $this->read_db_cluster-
>where('peserta_pindah_tugas.nisn', $nisn);
201.             } else {
202.                 $this->read_db_cluster-
>from('pendaftaran');
203.                 $this->read_db_cluster-
>join('peserta', 'peserta.id = pendaftaran.id_pes
erta');

```

```

204.         $this->read_db_cluster-
>where('peserta.nik', $nik);
205.         $this->read_db_cluster-
>where('peserta.nisn', $nisn);
206.     }
207.     } else {
208.         $this->read_db_cluster-
>select('*');
209.         $this->read_db_cluster-
>from('pendaftaran');
210.         $this->read_db_cluster-
>join('peserta', 'peserta.id = pendaftaran.id_pes
erta');
211.         $this->read_db_cluster-
>where('peserta.nik', $nik);
212.         $this->read_db_cluster-
>where('peserta.nisn', $nisn);
213.     }
214.
215.     return $this->read_db_cluster-
>get()->result();
216. }
217.
218.     public function update_status_pendafta
ran($id, $status, $komentar = null)
219.     {
220.         $array = array(
221.             'status' => $status,
222.             'komentar' => $komentar,
223.             'updated_at' => (new DateTime(
))->format('Y-m-d H:i:s')
224.         );
225.         $this->write_db_cluster-
>trans_start();
226.         $this->write_db_cluster-
>where('id', $id);
227.         $this->write_db_cluster-
>update('pendaftaran', $array);
228.         $this->write_db_cluster-
>trans_complete();
229.

```

```

230.         return $this->write_db_cluster-
>trans_status() === true;
231.     }
232.
233.     public function updateStatusPesertaPT(
    $id, $status, $komentar = null)
234.     {
235.         $array = array(
236.             'status' => $status,
237.             'komentar' => $komentar,
238.             'updated_at' => (new DateTime(
                ))->format('Y-m-d H:i:s')
239.
240.         );
241.         $this->write_db_cluster-
>trans_start();
242.         $this->write_db_cluster-
>where('id', $id);
243.         $this->write_db_cluster-
>update('peserta_pindah_tugas', $array);
244.         $this->write_db_cluster-
>trans_complete();
245.
246.         return $this->write_db_cluster-
>trans_status() === true;
247.     }
248.
249.     public function updateStatusPrestasi($
    idPrestasi)
250.     {
251.         $this->write_db_cluster-
>trans_start();
252.         $this->write_db_cluster-
>where('id', $idPrestasi);
253.         $this->write_db_cluster-
>update('prestasi', array('is_valid' => 1));
254.         $this->write_db_cluster-
>trans_complete();
255.
256.         if ($this->write_db_cluster-
>trans_status() === FALSE) {
257.             return False;

```

```

258.         } else {
259.             return True;
260.         }
261.     }
262.
263.     public function perbaruiNilaiTotal($id
Pendaftaran, $nilaiTotal)
264.     {
265.         $this->write_db_cluster-
>trans_start();
266.         $this->write_db_cluster-
>where('id_pendaftaran', $idPendaftaran);
267.         $this->write_db_cluster-
>update('pendaftaran_prestasi', array('poin_total
' => $nilaiTotal));
268.         $this->write_db_cluster-
>trans_complete();
269.
270.         return $this->write_db_cluster-
>trans_status() === TRUE;
271.     }
272.
273.     // END update status
274.
275.     public function sudahTerverifikasi($ni
k, $jalur, $idPeserta)
276.     {
277.         if ($jalur == JALUR_PERPINDAHAN_TU
GAS) {
278.             $this->read_db_cluster-
>select('peserta_pindah_tugas.status'); // Cek da
ri peserta_pindah_tugas bukan pendaftaran
279.             $this->read_db_cluster-
>from('peserta_pindah_tugas');
280.             $this->read_db_cluster-
>where('peserta_pindah_tugas.nik', $nik);
281.             $this->read_db_cluster-
>where('peserta_pindah_tugas.id', $idPeserta);
282.         } else {
283.             $this->read_db_cluster-
>select('pendaftaran.status, pendaftaran.tipe_pen
daftaran');

```

```

284.         $this->read_db_cluster-
>from('pendaftaran');
285.         $this->read_db_cluster-
>join('peserta', 'peserta.id = pendaftaran.id_pes
erta');
286.         $this->read_db_cluster-
>where('pendaftaran.tipe_pendaftaran', $jalur);
287.         $this->read_db_cluster-
>where('peserta.nik', $nik);
288.         $this->read_db_cluster-
>where('peserta.id', $idPeserta);
289.         }
290.         //$this->read_db_cluster-
>limit(1);
291.         $pendaftaran = $this-
>read_db_cluster->get()->result();
292.         return (!empty($pendaftaran)
&& $pendaftaran[0]-
>status == VERIFIKASI_DITERIMA);
293.     }
294.
295.         // Awal: bukti verifikasi
296.         public function getIdPendaftaranByNik(
$nik, $jalur = '')
297.         {
298.             if ($jalur == JALUR_PERPINDAHAN_TU
GAS) {
299.                 $this->read_db_cluster-
>select('id');
300.                 $this->read_db_cluster-
>from('peserta_pindah_tugas');
301.                 $this->read_db_cluster-
>where('nik', $nik);
302.                 $this->read_db_cluster-
>where('status', VERIFIKASI_DITERIMA);
303.             } else {
304.                 $this->read_db_cluster-
>select('pendaftaran.id');
305.                 $this->read_db_cluster-
>from('pendaftaran');

```

```

306.         $this->read_db_cluster-
>join("peserta", "pendaftaran.id_peserta = pesert
a.id");
307.         $this->read_db_cluster-
>where('pendaftaran.status', VERIFIKASI_DITERIMA)
;
308.     }
309.
310.         $this->read_db_cluster-
>limit(1);
311.         return $this->read_db_cluster-
>get()->result();
312.     }
313.
314.     public function getVerifikasi($id_pend
aftaran, $jalur = null)
315.     {
316.         if ($jalur == JALUR_PERPINDAHAN_TU
GAS) {
317.             $verifikasi = 'verifikasi_pind
ah_tugas';
318.             $peserta = 'peserta_pindah_tug
as';
319.
320.             $t1 = $verifikasi . '.id as id
_verifikasi,' . $verifikasi . '.waktu_verifikasi'
;
321.             $t2 = $peserta . '.*';
322.             $t3 = 'user_kantor.*';
323.
324.             $this->read_db_cluster-
>select($t1 . ', ' . $t2 . ', ' . $t3);
325.             $this->read_db_cluster-
>from($verifikasi);
326.             $this->read_db_cluster-
>where('id_peserta_pindah_tugas', $id_pendaftaran
);
327.             $this->read_db_cluster-
>order_by($verifikasi . '.id', 'desc');
328.             $this->read_db_cluster-
>join($peserta, $peserta . '.id = ' . $verifikasi
. '.id_peserta_pindah_tugas');

```



```

329.             $this->read_db_cluster-
>join('user_kantor', 'user_kantor.id = ' . $verifi
kasi . '.id_user');
330.             } else {
331.             $this->read_db_cluster-
>select('verifikasi.id as id_verifikasi, verifika
si.waktu_verifikasi, peserta.*, user_kantor.*');

332.             $this->read_db_cluster-
>from('verifikasi');
333.             $this->read_db_cluster-
>where('id_pendaftaran', $id_pendaftaran);
334.             $this->read_db_cluster-
>order_by('verifikasi.id', 'desc');
335.             $this->read_db_cluster-
>join('pendaftaran', 'pendaftaran.id = verifikasi
.id_pendaftaran');
336.             $this->read_db_cluster-
>join('peserta', 'peserta.id = pendaftaran.id_pes
erta');
337.             $this->read_db_cluster-
>join('user_kantor', 'user_kantor.id = verifikasi
.id_user');
338.             }
339.
340.             $this->read_db_cluster-
>limit(1);
341.             return $this->read_db_cluster-
>get()->result();
342.             }
343.
344.             // Akhir : bukti verifikasi
345.
346.             public function getDaftarPendaftaran($
jalur = '', $status = '')
347.             {
348.                 if ($jalur == JALUR_PERPINDAHAN_TU
GAS) {
349.                     $t1 = 'sekolah.nama_sekolah';

```

```

350.             $t2 = 'peserta_pindah_tugas.ni
k, peserta_pindah_tugas.nisn, peserta_pindah_tuga
s.nama, peserta_pindah_tugas.status';
351.             $t3 = 'verifikasi_pindah_tugas
.waktu_verifikasi, verifikasi_pindah_tugas.ketera
ngan';
352.
353.             $select = $t1 . ', ' . $t2;
354.
355.             if ($status != VERIFIKASI_BELU
M) {
356.                 $select = $select . ', ' .
$t3;
357.             }
358.
359.             $this->read_db_cluster-
>select($select);
360.             $this->read_db_cluster-
>from('peserta_pindah_tugas');
361.             $this->read_db_cluster-
>where('peserta_pindah_tugas.status', $status);
362.             $this->read_db_cluster-
>join('sekolah', 'sekolah.id = peserta_pindah_tug
as.id_sekolah');
363.
364.             if ($status != VERIFIKASI_BELU
M) {
365.                 $this->read_db_cluster-
>join('verifikasi_pindah_tugas', 'verifikasi_pind
ah_tugas.id_peserta_pindah_tugas = peserta_pindah
_tugas.id');
366.                 $this->read_db_cluster-
>where('verifikasi_pindah_tugas.aksi', $status);
367.             }
368.             } else {
369.                 $t1 = 'pendaftaran.tipe_pendaf
taran, pendaftaran.status';
370.                 $t2 = 'peserta.nik, peserta.ni
sn, peserta.nama';
371.                 $t3 = 'sekolah.nama_sekolah';

```

```

372.             $t4 = 'verifikasi.waktu_verifi
                kasi, verifikasi.keterangan';
373.
374.             $select = $t1 . ', ' . $t2 . '
                , ' . $t3;
375.
376.             if ($status != VERIFIKASI_BELU
                M) {
377.                 $select = $select . ', ' .
                $t4;
378.             }
379.
380.             $this->read_db_cluster-
                >select($select);
381.             $this->read_db_cluster-
                >from('pendaftaran');
382.             $this->read_db_cluster-
                >where('pendaftaran.status', $status);
383.             $this->read_db_cluster-
                >join('peserta', 'peserta.id = pendaftaran.id_pes
                erta');
384.             $this->read_db_cluster-
                >join('pilihan', 'pilihan.id_pendaftaran = pendaf
                taran.id and pilihan.indeks_pilihan = 1');
385.             $this->read_db_cluster-
                >join('sekolah', 'sekolah.id = pilihan.id_sekolah
                ');
386.
387.             if ($status != VERIFIKASI_BELU
                M) {
388.                 $this->read_db_cluster-
                >join('verifikasi', 'verifikasi.id_pendaftaran =
                pendaftaran.id');
389.                 $this->read_db_cluster-
                >where('verifikasi.aksi', $status);
390.             }
391.         }
392.
393.         return $this->read_db_cluster-
                >get()->result();
394.     }
395.

```

```

396.         // INSERT pendaftaran
397.         public function insertPendaftaran($dat
    as)
398.         {
399.             $this->write_db_cluster-
>trans_start();
400.             $this->write_db_cluster-
>insert('pendaftaran', $datas);
401.             $this->write_db_cluster-
>trans_complete();
402.
403.             return $this->write_db_cluster-
>trans_status() === true;
404.         }
405.
406.         public function insertPilihan($datas)
407.         {
408.             $this->write_db_cluster-
>trans_start();
409.             $this->write_db_cluster-
>insert('pilihan', $datas);
410.             $this->write_db_cluster-
>trans_complete();
411.
412.             return $this->write_db_cluster-
>trans_status() === true;
413.         }
414.
415.         public function deletePendaftaranByPes
    erta($idPeserta)
416.         {
417.             $this->write_db_cluster-
>trans_start();
418.             $this->write_db_cluster-
>where('id_peserta', $idPeserta);
419.             $this->write_db_cluster-
>delete('pendaftaran');
420.             $this->write_db_cluster-
>trans_complete();
421.

```

```

422.         return $this->write_db_cluster-
         >trans_status() === true;
423.     }
424.
425.         // Sudah ditolak
426.         public function sudahDitolak($idPendaftaran)
427.         {
428.             $this->read_db_cluster-
         >select('status');
429.             $this->read_db_cluster-
         >from('pendaftaran');
430.             $this->read_db_cluster-
         >where('pendaftaran.id', $idPendaftaran);
431.
432.             $hasil = $this->read_db_cluster-
         >get()->result();
433.             return count($hasil) > 0 && $hasil
         [0]->status == VERIFIKASI_DITOLAK;
434.         }
435.
436.         public function sudahDitolakPT($idPesertaPT)
437.         {
438.             $this->read_db_cluster-
         >select('status');
439.             $this->read_db_cluster-
         >from('peserta_pindah_tugas');
440.             $this->read_db_cluster-
         >where('peserta_pindah_tugas.id', $idPesertaPT);
441.
442.             $hasil = $this->read_db_cluster-
         >get()->result();
443.             return count($hasil) > 0 && $hasil
         [0]->status == VERIFIKASI_DITOLAK;
444.         }
445.
446.         // Sudah disetujui
447.         public function sudahDisetujuiPT($nisn
         , $nik)
448.         {

```

```

449.         $this->read_db_cluster-
>select('status');
450.         $this->read_db_cluster-
>from('peserta_pindah_tugas');
451.         $this->read_db_cluster-
>where('nisp', $nisp);
452.         $this->read_db_cluster-
>where('nik', $nik);
453.         //         $this->read_db_cluster-
>limit(1);
454.         $this->read_db_cluster-
>order_by('created_at', 'DESC');
455.
456.         $hasil = $this->read_db_cluster-
>get()->result();
457.         return count($hasil) > 0 && $hasil
[0]->status == VERIFIKASI_DITERIMA;
458.     }
459.
460.         //VISUAL
461.         /*public function countPendaftaran($ja
lur, $status){
462.             $this->read_db_cluster-
>select('pendaftaran.id');
463.             $this->read_db_cluster-
>from('pendaftaran');
464.             $this->read_db_cluster-
>where('pendaftaran.tipe_pendaftaran', $jalur);
465.             $this->read_db_cluster-
>where('pendaftaran.status', $status);
466.
467.             return count($this-
>read_db_cluster->get()->result());
468.         }*/
469.
470.         public function countPendaftaran($jalu
r, $status)
471.         {
472.             if (in_array($jalur, [JALUR_AFIRMA
SI, JALUR_PRESTASI])) {
473.                 $this->read_db_cluster-
>select('id');

```

```

474.         $this->read_db_cluster-
>from('pendaftaran');
475.         $this->read_db_cluster-
>where('tipe_pendaftaran', $jalur);
476.         $this->read_db_cluster-
>where('status', $status);
477.         } elseif ($jalur === JALUR_ZONASI
|| $jalur === JALUR_TAHAP3) {
478.         $this->read_db_cluster-
>select('id');
479.         $this->read_db_cluster-
>from('pendaftaran');
480.         $this->read_db_cluster-
>where('tipe_pendaftaran', $jalur);
481.
482.         if ($status === VERIFIKASI_DIT
ERIMA) {
483.         $this->read_db_cluster-
>where('status', ZONASI_DITERIMA);
484.         } else {
485.         $this->read_db_cluster-
>where('status', ZONASI_DITOLAK);
486.         }
487.         } else {
488.         $this->read_db_cluster-
>select('peserta_pindah_tugas.id as pendaftaran_i
d');
489.         $this->read_db_cluster-
>from('peserta_pindah_tugas');
490.         $this->read_db_cluster-
>where('status', $status);
491.         }
492.
493.         return count($this-
>read_db_cluster->get()->result());
494.         }
495.
496.         public function getPendaftaranPerDay($
jalur, $idSekolah = null)
497.         {
498.         if (isset($idSekolah)) {

```

```

499.             $query = sprintf("select count
(1) total, date(created_at) tgl_pendaftaran
500.             from pendaftaran
501.             join peserta p on pen
daftaran.id_peserta = p.id
502.             join pilihan p2 on pe
ndaftaran.id = p2.id_pendaftaran
503.             where date(created_at) between
date('2020-06-17') and date('2020-06-28')
504.             and year(date(created_at)) !
= 0000
505.             and tipe_pendaftaran = '%s'

506.             and p2.id_sekolah = '%s'
507.             group by date(created_at)
508.             order by date(created_at);", $
jalur, $idSekolah);
509.         } else {
510.             $query = sprintf("select count
(1) total, date(created_at) tgl_pendaftaran
511.             from pendaftaran
512.             join peserta p on pen
daftaran.id_peserta = p.id
513.             join pilihan p2 on pe
ndaftaran.id = p2.id_pendaftaran
514.             where date(created_at) between
date('2020-06-17') and date('2020-06-28')
515.             and year(date(created_at)) !
= 0000
516.             and tipe_pendaftaran = '%s'

517.             group by date(created_at)
518.             order by date(created_at);", $
jalur);
519.         }
520.
521.         return $this->read_db_cluster-
>query($query)->result();
522.     }
523.
524. }

```


5.2.1.3. Model Siswa

Tabel 5.2-3 Model Siswa mengenai informasi pendaftaran dan pendataan

```
1. <?php
2.
3. class Siswa extends CI_Model
4. {
5.
6.     public function getSiswaVerifikasiData($jalur
7.     , $status)
8.     {
9.         if ($jalur == 'prestasi') {
10.            $this->read_db_cluster-
11.            >select('*');
12.            $this->read_db_cluster-
13.            >from('pendaftaran');
14.            $this->read_db_cluster-
15.            >where('pendaftaran.status', $status);
16.            $this->read_db_cluster-
17.            >where('pendaftaran.tipe_pendaftaran', $jalur);
18.            $this->read_db_cluster-
19.            >join("peserta", "pendaftaran.id_peserta = pesert
20.            a.id");
21.            $this->read_db_cluster-
22.            >join("pendaftaran_prestasi", "pendaftaran_presta
23.            si.id_pendaftaran = pendaftaran.id");
24.        } else {
25.            $this->read_db_cluster-
26.            >select('*');
27.            $this->read_db_cluster-
28.            >from('pendaftaran');
29.            $this->read_db_cluster-
30.            >where('pendaftaran.status', $status);
31.            $this->read_db_cluster-
32.            >where('pendaftaran.tipe_pendaftaran', $jalur);
33.            $this->read_db_cluster-
34.            >join("peserta", "pendaftaran.id_peserta = pesert
35.            a.id");
```

```

22.     }
23.     return $this->read_db_cluster->get()-
>result();
24. }
25.
26. public function get_daftar_siswa_pendataan($s
umber_data, $status, $sekolahId = '')
27. {
28.     $this->read_db_cluster-
>select('peserta_pendataan.id as id_peserta_penda
taan, peserta_pendataan.*');
29.     $this->read_db_cluster-
>from('peserta_pendataan');
30.     $this->read_db_cluster-
>where('peserta_pendataan.sumber_data', $sumber_d
ata);
31.     $this->read_db_cluster-
>where('peserta_pendataan.status', $status);
32.
33.     if (!empty($sekolahId)) {
34.         $this->read_db_cluster-
>where('peserta_pendataan.id_sekolah_verifikator'
, $sekolahId);
35.     }
36.
37.     return $this->read_db_cluster->get()-
>result();
38.     //     $query = "select pd.* from pesert
a_pendataan pd
39.     //         where pd.nik not in (select p
.nik from peserta p)
40.     //         and pd.nisn not in (select p.
nisn from peserta p)
41.     //         and pd.status=''' . $status .
'''' and pd.sumber_data=''' . $sumber_data . '''';
42.
43.     //     return $this->read_db_cluster-
>query($query)->result();
44. }
45.
46. public function getByNikNisn($nik, $nisn, $se
lect = null)

```

```

47.     {
48.         if (!emptyempty($select)) {
49.             $this->read_db_cluster-
>select($select);
50.         } else {
51.             $this->read_db_cluster-
>select('*');
52.         }
53.         $this->read_db_cluster-
>from('peserta_pendataan');
54.         $this->read_db_cluster-
>where('nik', $nik);
55.         $this->read_db_cluster-
>where('nispn', $nispn);
56.         return $this->read_db_cluster->get()-
>result();
57.     }
58.
59.     public function adaDiPeserta($nispn = '')
60.     {
61.         $this->read_db_cluster->select('id');
62.         $this->read_db_cluster-
>from('peserta');
63.         $this->read_db_cluster-
>where('nispn', $nispn);
64.         $this->read_db_cluster->limit(1);
65.
66.         $hasil = $this->read_db_cluster->get()-
>result();
67.         return count($hasil) > 0;
68.     }
69.
70.     public function sudahDitolak($id)
71.     {
72.         $this->read_db_cluster-
>select('status');
73.         $this->read_db_cluster-
>from('peserta_pendataan');
74.         $this->read_db_cluster-
>where('peserta_pendataan.id', $id);
75.         $this->read_db_cluster->limit(1);
76.

```

```

77.     $hasil = $this->read_db_cluster->get()-
>result();
78.     return count($hasil) > 0 && $hasil[0]-
>status == VERIFIKASI_DITOLAK;
79.     }
80.
81.     public function get_siswa_pendataan($nik, $id
)
82.     {
83.         $this->read_db_cluster-
>select('peserta_pendataan.id as id_peserta_penda
taan, peserta_pendataan.*, peserta_pendataan.nama
_kelurahan, peserta_pendataan.nama_kecamatan');
84.         $this->read_db_cluster-
>from('peserta_pendataan');
85.         $this->read_db_cluster-
>where('peserta_pendataan.nik', $nik);
86.         $this->read_db_cluster-
>where('peserta_pendataan.id', $id);
87.
88.         $this->read_db_cluster-
>order_by('peserta_pendataan.created_at', 'DESC')
;
89.
90.         return $this->read_db_cluster->get()-
>result();
91.     }
92.
93.     public function getPendataanById($id)
94.     {
95.         $this->read_db_cluster-
>select('peserta_pendataan.*, kelurahan.nama_kelu
rahan, kecamatan.nama_kecamatan');
96.         $this->read_db_cluster-
>from('peserta_pendataan');
97.         $this->read_db_cluster-
>where('peserta_pendataan.id', $id);
98.         $this->read_db_cluster-
>join('kelurahan', "peserta_pendataan.id_keluraha
n = kelurahan.id");

```

```

99.         $this->read_db_cluster-
>join('kecamatan', "kelurahan.id_kecamatan = keca
matan.id");
100.
101.         $this->read_db_cluster-
>order_by('peserta_pendataan.created_at', 'DESC')
;
102.
103.         return $this->read_db_cluster-
>get()->result();
104.     }
105.
106.     public function getPesertaByNisn($nisn
)
107.     {
108.         $this->read_db_cluster-
>select('peserta.*, kelurahan.nama_kelurahan, kec
amatan.nama_kecamatan');
109.         $this->read_db_cluster-
>from('peserta');
110.         $this->read_db_cluster-
>where('peserta.nisn', $nisn);
111.         $this->read_db_cluster-
>join('kelurahan', "peserta.id_kelurahan = kelura
han.id");
112.         $this->read_db_cluster-
>join('kecamatan', "kelurahan.id_kecamatan = keca
matan.id");
113.         $this->read_db_cluster-
>limit(1);
114.
115.         return $this->read_db_cluster-
>get()->result();
116.     }
117.
118.     public function isPendataanVerified($i
d)
119.     {
120.         $this->read_db_cluster-
>select('status');
121.         $this->read_db_cluster-
>from('peserta_pendataan');

```

```

122.         $this->read_db_cluster-
>where('id', $id);
123.         $this->read_db_cluster-
>order_by('created_at', 'DESC');
124.         return $this->read_db_cluster-
>get()->result()[0];
125.     }
126.
127.         // END get data siswa
128.
129.         public function update_status_siswa_pe
ndataan($id, $status, $komentar = null)
130.     {
131.         $array = array(
132.             'status' => $status,
133.             'komentar' => $komentar,
134.             'updated_at' => (new DateTime(
))->format('Y-m-d H:i:s')
135.         );
136.
137.         $this->write_db_cluster-
>trans_start();
138.         $this->write_db_cluster-
>where('id', $id);
139.         $this->write_db_cluster-
>update('peserta_pendaftaran', $array);
140.         $this->write_db_cluster-
>trans_complete();
141.
142.         if ($this->write_db_cluster-
>trans_status() === FALSE) {
143.             return False;
144.         } else {
145.             return True;
146.         }
147.     }
148.
149.         public function insertPendaftaranToPeser
ta($datas)
150.     {
151.         $this->write_db_cluster-
>trans_start();

```

```

152.         $this->write_db_cluster-
>insert('peserta', $datas);
153.         $this->write_db_cluster-
>trans_complete();
154.
155.         if ($this->write_db_cluster-
>trans_status() === true) {
156.             return $this-
>getPesertaByNisn($datas['nisn']);
157.         } else {
158.             return false;
159.         }
160.     }
161.
162.     public function deleteSiswaPendataan($
id, $status)
163.     {
164.         $this->write_db_cluster-
>trans_start();
165.         $this->write_db_cluster-
>delete('peserta_pendataan', array('id' => $id, '
status' => $status));
166.         $this->write_db_cluster-
>trans_complete();
167.
168.         return $this->write_db_cluster-
>trans_status() === true;
169.     }
170.
171.     public function deletePeserta($id, $st
atus)
172.     {
173.         $this->write_db_cluster-
>trans_start();
174.         $this->write_db_cluster-
>where('id', $id);
175.         $this->write_db_cluster-
>where('status', $status);
176.         $this->write_db_cluster-
>delete('peserta');
177.         $this->write_db_cluster-
>trans_complete();

```

```

178.
179.         return $this->write_db_cluster-
180.             >trans_status() === true;
181.     }
182.     public function getSiswaByIdPendaftara
183.         n($idPendaftaran)
184.     {
185.         $this->read_db_cluster-
186.             >select("pendaftaran.id_peserta, peserta.nama, pe
187.                 ndaftaran.tipe_pendaftaran");
188.         $this->read_db_cluster-
189.             >where('pendaftaran.id', $idPendaftaran);
190.         $this->read_db_cluster-
191.             >from('pendaftaran');
192.         $this->read_db_cluster-
193.             >join('peserta', 'peserta.id = pendaftaran.id_pes
194.                 erta');
195.         return $this->read_db_cluster-
196.             >get()->result();
197.     }
198.     public function getVerifikasi($id_pese
199.         rta_pendataan)
200.     {
201.         $this->read_db_cluster-
202.             >select('verifikasi_pendataan.id as id_verifikasi
203.                 , verifikasi_pendataan.waktu_verifikasi, peserta_
204.                 pendataan.*, user_kantor.*');
205.         $this->read_db_cluster-
206.             >from('verifikasi_pendataan');
207.         $this->read_db_cluster-
208.             >where('verifikasi_pendataan.id_peserta_pendataan
209.                 ', $id_peserta_pendataan);
210.         $this->read_db_cluster-
211.             >join('peserta_pendataan', 'peserta_pendataan.id
212.                 = verifikasi_pendataan.id_peserta_pendataan');
213.         $this->read_db_cluster-
214.             >where('peserta_pendataan.status', VERIFIKASI_DIT
215.                 ERIMA);

```



```

198.         $this->read_db_cluster-
>join('user_kantor', 'user_kantor.id = verifikasi
   _pendataan.id_user');
199.         $this->read_db_cluster-
>order_by('verifikasi_pendataan.id', 'desc');
200.         $this->read_db_cluster-
>limit(1);
201.
202.         return $this->read_db_cluster-
>get()->result();
203.     }
204.
205.     public function getDaftarPendataan($st
   atus = '')
206.     {
207.         $t1 = 'peserta_pendataan.nik, pese
   rta_pendataan.nama, peserta_pendataan.sumber_data
   , peserta_pendataan.status';
208.         $t2 = 'sekolah.nama_sekolah';
209.         $t3 = 'verifikasi_pendataan.waktu_
   verifikasi, verifikasi_pendataan.keterangan';
210.         $select = $t1 . ', ' . $t2;
211.
212.         if ($status != VERIFIKASI_BELUM) {
213.             $select = $select . ', ' . $t3
   ;
214.         }
215.
216.         $this->read_db_cluster-
>select($select);
217.         $this->read_db_cluster-
>from('peserta_pendataan');
218.         $this->read_db_cluster-
>join('sekolah', 'sekolah.id = peserta_pendataan.
   id_sekolah_verifikator');
219.         $this->read_db_cluster-
>where('peserta_pendataan.status', $status);
220.
221.         if ($status != VERIFIKASI_BELUM) {

```

```

222.             $this->read_db_cluster-
>join('verifikasi_pendataan', 'verifikasi_pendata
an.id_peserta_pendataan = peserta_pendataan.id');
223.             }
224.
225.             return $this->read_db_cluster-
>get()->result();
226.         }
227.
228.         public function getDafulTahap3ByPendaft
aran($idPendaftaran)
229.         {
230.             $query = sprintf("select peserta_t
ahap3.attachment
231.                             from peserta_tahap3
232.                             join peserta p on pes
erta_tahap3.nisn = p.nisn
233.                             join pendaftaran p2 o
n p.id = p2.id_peserta
234.                             where p2.id = %s;", $idPendaft
aran);
235.
236.             return $this->read_db_cluster-
>query($query)->result();
237.         }
238.
239.         // VISUAL
240.         public function countPesertaPendataan(
$jalur, $status)
241.         {
242.             //             $this->read_db_cluster-
>start_cache();
243.             $this->read_db_cluster-
>select('peserta_pendataan.id as id_peserta_penda
taan');
244.             $this->read_db_cluster-
>from('peserta_pendataan');
245.             $this->read_db_cluster-
>where('peserta_pendataan.sumber_data', $jalur);

```

```

246.         $this->read_db_cluster-
>where('peserta_pendataan.status', $status);
247.         //         $this->read_db_cluster-
>stop_cache());
248.
249.         return count($this-
>read_db_cluster->get()->result());
250.     }
251. }

```

5.2.1.4. Model Daftar Ulang

Tabel 5.2-4 Model Daftar Ulang Peserta yang diterima

```

1. <?php
2.
3.
4. class DaftarUlang_model extends CI_Model
5. {
6.     public function insertDaftarUlang($datas)
7.     {
8.         $this->write_db_cluster->trans_start();
9.         $this->write_db_cluster-
>insert('daftar_ulang', $datas);
10.        $this->write_db_cluster-
>trans_complete();
11.
12.        return $this->write_db_cluster-
>trans_status === true;
13.    }
14.
15.    public function updateStatusPendaftaran($idPe
ndaftaran)
16.    {
17.        $this->write_db_cluster->trans_start();
18.        $this->write_db_cluster-
>where('id', $idPendaftaran);
19.        $this->write_db_cluster-
>update('pendaftaran', array('status_daftar_ulang
' => 1));

```

```

20.         $this->write_db_cluster-
>trans_complete();
21.
22.         return $this->write_db_cluster-
>trans_status === true;
23.     }
24.
25.     public function isSiswaExist($idPendaftaran)
26.     {
27.         $this->read_db_cluster->select('nisn');
28.         $this->read_db_cluster-
>from('daftar_ulang');
29.         $this->read_db_cluster-
>where('id_pendaftaran', $idPendaftaran);
30.
31.         $res = $this->read_db_cluster->get()-
>result();
32.
33.         return count($res) > 0;
34.     }
35.
36.     public function getDaftarUlang($jalur, $status, $idSekolah = '')
37.     {
38.         $this->read_db_cluster-
>select('daftar_ulang.*, peserta.asal_sekolah, pe
serta.nik, peserta.nisn');
39.         $this->read_db_cluster-
>from('daftar_ulang');
40.         $this->read_db_cluster-
>join('peserta', 'peserta.id = daftar_ulang.id_pe
serta');
41.         $this->read_db_cluster-
>where('daftar_ulang.status', $status);
42.         $this->read_db_cluster-
>where('daftar_ulang.jalur', $jalur);
43.
44.         if (isset($idSekolah) && !empty($idS
ekolah)) {
45.             $this->read_db_cluster-
>where('id_sekolah', $idSekolah);

```

```

46.     }
47.
48.     $this->read_db_cluster-
>order_by('daftar_ulang.nama_peserta');
49.     return $this->read_db_cluster->get()-
>result();
50.     }
51.
52.     public function getConfigRank($param)
53.     {
54.         $this->output_db_ranking-
>select('value');
55.         $this->output_db_ranking-
>from('configs');
56.         $this->output_db_ranking-
>where('param', $param);
57.
58.         return $this->output_db_ranking->get()-
>result()[0]->value;
59.     }
60.
61.     // Utamanya untuk yang belum daful
62.     public function getOutputRanking($jalur, $idS
ekolah = '', $status_daftar_ulang = DAFTAR_ULANG_
BELUM)
63.     {
64.         switch ($jalur) {
65.             case JALUR_AFIRMASI:
66.                 $table = 'output_afirmasi' . $thi
s->getConfigRank(JALUR_AFIRMASI);
67.                 break;
68.             case JALUR_PERPINDAHAN_TUGAS:
69.                 $table = 'output_pindah_tugas' .
$this->getConfigRank(JALUR_PERPINDAHAN_TUGAS);
70.                 break;
71.             case JALUR_PRESTASI:
72.                 $table = 'output_prestasi' . $thi
s->getConfigRank(JALUR_PRESTASI);
73.                 break;
74.             case JALUR_TAHAP3:
75.                 $table = 'output_tahap3' . $this-
>getConfigRank(JALUR_TAHAP3);

```

```

76.             break;
77.             default:
78.                 $table = 'output_zonasi' . $this-
>getConfigRank(JALUR_ZONASI);
79.                 break;
80.
81.         }
82.         $this->output_db_ranking->select('*');
83.         $this->output_db_ranking->from($table);
84.
85.         if (isset($idSekolah)) {
86.             $this->output_db_ranking-
>where('id_sekolah', $idSekolah);
87.         }
88.
89.         return $this->output_db_ranking->get()-
>result();
90.     }
91.
92.     public function getStatusDaftarUlang($idPenda
ftaran, $jalur)
93.     {
94.         $this->read_db_cluster-
>select('status_daftar_ulang');
95.         $this->read_db_cluster-
>from('pendaftaran');
96.         $this->read_db_cluster-
>where('pendaftaran.id', $idPendaftaran);
97.         $this->read_db_cluster-
>where('pendaftaran.tipe_pendaftaran', $jalur);
98.
99.
100.        return (int)$this-
>read_db_cluster->get()->result()[0]-
>status_daftar_ulang;
101.    }
102.
103.    public function getSiswaInRanking($nis
n, $idOutput, $jalur)
104.    {
105.        switch ($jalur) {
106.            case JALUR_AFIRMASI:

```

```

107.         $table = 'output_afirmasi'
    . $this->getConfigRank(JALUR_AFIRMASI);
108.         break;
109.         case JALUR_PERPINDAHAN_TUGAS:

110.         $table = 'output_pindah_tu
gas' . $this-
>getConfigRank(JALUR_PERPINDAHAN_TUGAS);
111.         break;
112.         case JALUR_PRESTASI:
113.         $table = 'output_prestasi'
    . $this->getConfigRank(JALUR_PRESTASI);
114.         break;
115.         case JALUR_TAHAP3:
116.         $table = 'output_tahap3' .
    $this->getConfigRank(JALUR_TAHAP3);
117.         break;
118.         default:
119.         $table = 'output_zonasi' .
    $this->getConfigRank(JALUR_ZONASI);
120.         break;
121.
122.     }
123.
124.     $this->output_db_ranking-
>select('*');
125.     $this->output_db_ranking-
>from($table);
126.     $this->output_db_ranking-
>where('nisn', $nisn);
127.     $this->output_db_ranking-
>where('id', $idOutput);
128.
129.     return $this->output_db_ranking-
>get()->result()[0];
130.     }
131.
132.     public function getSiswaInDaful($nisn,
    $idPendaftaran, $jalur)
133.     {
134.         $this->read_db_cluster-
>select('*');

```

```

135.         $this->read_db_cluster-
>from('daftar_ulang');
136.         $this->read_db_cluster-
>where('id_pendaftaran', $idPendaftaran);
137.         $this->read_db_cluster-
>where('nisn', $nisn);
138.         $this->read_db_cluster-
>where('jalur', $jalur);
139.
140.         return $this->read_db_cluster-
>get()->result()[0];
141.     }
142.
143.     public function getListPrestasiFromOut
put($idPendaftaran)
144.     {
145.         $this->read_db_cluster-
>select('*');
146.         $this->read_db_cluster-
>from('prestasi');
147.         $this->read_db_cluster-
>join('pendaftaran_prestasi', 'pendaftaran_presta
si.id = prestasi.id_pendaftaran_prestasi');
148.         $this->read_db_cluster-
>where('pendaftaran_prestasi.id_pendaftaran', $id
Pendaftaran);
149.
150.         return $this->read_db_cluster-
>get()->result();
151.     }
152.
153.     // Copy dari Pendaftaran.php dengan pe
nyesuaian
154.     public function getSiswaByNisn(
$nisn, $jalur, $idPendaftaran)
155.     {
156.         if ($jalur == JALUR_PRESTASI) {
157.             $this->read_db_cluster-
>select('pendaftaran.id as pendaftaran_id, pesert
a.*',

```



```

158.         pendaftaran_prestasi.id as pen
    daftaran_prestasi_id, pendaftaran_prestasi.poin_t
    otal');
159.         $this->read_db_cluster-
    >from('peserta');
160.         $this->read_db_cluster-
    >join("pendaftaran", "pendaftaran.id_peserta = pe
    serta.id");
161.         $this->read_db_cluster-
    >join("pendaftaran_prestasi", "pendaftaran_presta
    si.id_pendaftaran = pendaftaran.id");
162.         $this->read_db_cluster-
    >where('peserta.nisn', $nisn);
163.         //$this->read_db_cluster-
    >limit(1);
164.         $this->read_db_cluster-
    >where('pendaftaran.id', $idPendaftaran);
165.         } // Afirmasi
166.         else {
167.         $this->read_db_cluster-
    >select('pendaftaran.id as pendaftaran_id, pesert
    a.*');
168.         $this->read_db_cluster-
    >from('pendaftaran');
169.         $this->read_db_cluster-
    >join("peserta", "peserta.id = pendaftaran.id_pes
    erta");
170.         $this->read_db_cluster-
    >where('pendaftaran.id', $idPendaftaran);
171.         //$this->read_db_cluster-
    >limit(1);
172.         $this->read_db_cluster-
    >where('peserta.nisn', $nisn);
173.         }
174.         // Double cek aja
175.         $this->read_db_cluster-
    >where('pendaftaran.tipe_pendaftaran', $jalur);
176.         if ($jalur === JALUR_ZONASI || $ja
    lur === JALUR_TAHAP3) {
177.         $this->read_db_cluster-
    >where('pendaftaran.status', ZONASI_DITERIMA);
178.         } else {

```

```

179.             $this->read_db_cluster-
>where('pendaftaran.status', VERIFIKASI_DITERIMA)
;
180.         }
181.
182.         // Kalau double
183.         $this->read_db_cluster-
>order_by('pendaftaran.created_at', 'DESC');
184.
185.         return $this->read_db_cluster-
>get()->result()[0];
186.     }
187.
188.     public function getSiswaPindahTugas($n
isn)
189.     {
190.         $this->read_db_cluster-
>select('peserta_pindah_tugas.id as id_peserta_pi
ndah_tugas, peserta_pindah_tugas.*');
191.         $this->read_db_cluster-
>from('peserta_pindah_tugas');
192.         $this->read_db_cluster-
>where('peserta_pindah_tugas.nisn', $nisn);
193.         $this->read_db_cluster-
>where('peserta_pindah_tugas.status', VERIFIKASI_
DITERIMA);
194.         $this->read_db_cluster-
>where('peserta_pindah_tugas.is_checked', VERIFIK
ASI_DITERIMA);
195.
196.         // Kalau double
197.         $this->read_db_cluster-
>order_by('peserta_pindah_tugas.created_at', 'DES
C');
198.
199.         return $this->read_db_cluster-
>get()->result()[0];
200.     }
201.
202.     public function getSiswaTahap3($nisn)
203.     {

```

```

204.         $this->read_db_cluster-
>select('peserta_tahap3.*');
205.         $this->read_db_cluster-
>from('peserta_tahap3');
206.         $this->read_db_cluster-
>where('peserta_tahap3.status', VERIFIKASI_DITERI
MA);
207.         $this->read_db_cluster-
>where('peserta_tahap3.nisn', $nisn);
208.
209.         return $this->read_db_cluster-
>get()->result()[0];
210.     }
211.
212.     public function getDaftarUlangPerDay($
dateTime, $sekolahId)
213.     {
214.         $this->read_db_cluster-
>select('*');
215.         $this->read_db_cluster-
>from('daftar_ulang');
216.         $this->read_db_cluster-
>where('');
217.     }
218.     }

```

5.2.2. Aplikasi Pendaftaran

5.2.2.1. Model Autentifikasi

Tabel 5.2-5 Tabel Autentikasi Aplikasi Pendaftaran

```
1. <?php
2.
3. namespace App\Core\Domain\Model;
4.
5. use Illuminate\Foundation\Auth\User as Authenticatable;
6.
7. /**
8.  * @property-read int $id
9.  * @property string $username
10. * @property string $password
11. * @property int $id_peserta
12. * @property int $flag
13. * @property-read Peserta $peserta
14. */
15. class AuthPeserta extends Authenticatable
16. {
17.     public const DEFAULT = 0;
18.     public const SUDAH_DITERIMA = 1;
19.     public const UMUR_MAKSIMUM = 2;
20.
21.     protected $table = 'auth_peserta';
22.     public $timestamps = false;
23.
24.     /**
25.      * The attributes that are mass assignable.
26.      *
27.      * @var array
28.      */
29.     protected $fillable = [
30.         'username', 'password', 'id_peserta', 'flag'
31.     ];
32.
33.     /**
```

```

34.     * The attributes that should be hidden for a
      rrays.
35.     *
36.     * @var array
37.     */
38.     protected $hidden = [
39.         'password', 'remember_token',
40.     ];
41.
42.     public function peserta()
43.     {
44.         return $this->
>belongsTo(Peserta::class, 'id_peserta', 'id');
45.     }
46. }

```

5.2.2.2. Model Daftar Ulang

Tabel 5.2-6 Model Daftar Ulang Aplikasi Pendaftaran

```

1. <?php
2.
3. namespace App\Core\Domain\Model;
4.
5. use DateTime;
6. use Illuminate\Database\Eloquent\Model;
7.
8. class DaftarUlang extends Model
9. {
10.     protected $table = 'daftar_ulang';
11.
12.     protected $fillable = [
13.         'id_sekolah',
14.         'id_pendaftaran',
15.         'id_peserta',
16.         'indeks_peringkat',
17.         'nispn',
18.         'nama_peserta',
19.         'jalur',

```

```

20.         'tgl_lahir',
21.         'nama_sekolah',
22.         'indeks_pilihan',
23.         'jarak',
24.         'poin',
25.         'status',
26.         'no_hp'
27.     ];
28. }

```

5.2.2.3. Model Pendaftaran

Tabel 5.2-7 Model Pendaftaran

```

1. <?php
2.
3. namespace App\Core\Domain\Model;
4.
5. use Illuminate\Database\Eloquent\Model;
6.
7. /**
8.  * @property-read int $id
9.  * @property int $id_peserta
10.  * @property string $tipe_pendaftaran
11.  * @property int $status
12.  * @property bool $uploaded_berkas
13.  */
14. class Pendaftaran extends Model
15. {
16.     public const BARU_DIBUAT = -2;
17.     public const DITOLAK = -1;
18.     public const PENDING_VERIFIKASI = 0;
19.     public const DITERIMA = 1;
20.     public const VERIFIKASI_ULANG = 2;
21.     public const PERMANEN = 3; // Pendaftaran per
    manen setelah diterima verifikator dan pilihan se
    kolah sudah permanen
22.
23.     public const JALUR_ZONASI = 'zonasi';
24.     public const JALUR_PRESTASI = 'prestasi';

```

```

25.     public const JALUR_AFIRMASI      = 'afirmasi';
26.     public const JALUR_PINDAH_TUGAS = 'pindah_tugas';
27.     public const JALUR_TAHAP3       = 'tahap3';
28.
29.     protected $table = 'pendaftaran';
30.     public $timestamps = true;
31.
32.     protected $fillable = [
33.         'id_peserta',
34.         'tipe_pendaftaran',
35.         'status',
36.         'uploaded_berkas'
37.     ];
38.
39.     protected $attributes = [
40.         'status' => self::BARU_DIBUAT
41.     ];
42.
43.     public function pilihans()
44.     {
45.         return $this->hasMany(Pilihan::class, 'id_pendaftaran', 'id');
46.     }
47.
48.     public function pendaftaran_prestasi()
49.     {
50.         return $this->hasOne(PendaftaranPrestasi::class, 'id_pendaftaran', 'id');
51.     }
52. }

```

5.2.2.4. Model Pendaftaran Prestasi

Tabel 5.2-8 Model Pendaftaran Jalur Prestasi

```
1. <?php
```

```

2.
3. namespace App\Core\Domain\Model;
4.
5. use Illuminate\Database\Eloquent\Model;
6.
7. /**
8.  * @property-read int $id
9.  * @property int $id_pendaftaran
10. * @property int $poin_total
11. */
12. class PendaftaranPrestasi extends Model
13. {
14.     protected $table = 'pendaftaran_prestasi';
15.     public $timestamps = false;
16.
17.     protected $fillable = [
18.         'id_pendaftaran',
19.         'poin_total'
20.     ];
21. }

```

5.2.2.5. Model Peserta

Tabel 5.2-9 Model Peserta Pendaftaran

```

1. <?php
2.
3. namespace App\Core\Domain\Model;
4.
5. use DateTime;
6. use Illuminate\Database\Eloquent\Model;
7.
8. /**
9.  * @property-read int $id
10. * @property string $nama
11. * @property string $nisn
12. * @property string $nik
13. * @property string $asal_sekolah
14. * @property DateTime $tgl_lahir
15. * @property string $tempat_lahir

```



```

16. * @property string $jenis_kelamin
17. * @property string $nama_ortu
18. * @property int $rt
19. * @property int $rw
20. * @property int $id_kelurahan
21. * @property string $nama_kelurahan
22. * @property string $nama_kecamatan
23. * @property string $alamat
24. * @property float $latitude
25. * @property float $longitude
26. * @property string $sumber_data
27. * @property int $status
28. * @property bool $uploaded_berkas
29. */
30. class Peserta extends Model
31. {
32.     public const BELUM_PERMANEN = -2;
33.     public const DITOLAK = -1;
34.     public const BELUM_VERIFIKASI = 0;
35.     public const DITERIMA = 1;
36.     public const VERIFIKASI_ULANG = 2;
37.
38.     public const SUMBER_MASTER = 'master';
39.     public const SUMBER_LUAR_KOTA = 'luar_kota';
40.
41.     public const SUMBER_TAHUN_LALU = 'tahun_lalu'
42.     ;
43.     public const SUMBER_SD_MI = 'sd_mi';
44.     public const SUMBER_PINDAH_TUGAS = 'p_tugas';
45.
46.     public const SUMBER_TAHAP3 = 'tahap3';
47.
48.     public const KELURAHAN_LUAR_KOTA = 999;
49.
50.     protected $table = 'peserta';
51.     public $timestamps = false;
52.
53.     protected $fillable = [
54.         'nama',
55.         'nisp',
56.         'nik',
57.         'asal_sekolah',

```

```

55.     'tgl_lahir',
56.     'tempat_lahir',
57.     'jenis_kelamin',
58.     'nama_ortu',
59.     'rt',
60.     'rw',
61.     'id_kelurahan',
62.     'nama_kelurahan',
63.     'nama_kecamatan',
64.     'alamat',
65.     'latitude',
66.     'longitude',
67.     'sumber_data',
68.     'status',
69.     'uploaded_berkas'
70. ];
71.
72.     public function getTglLahirAttribute($value)
73.     {
74.         return new DateTime($value);
75.     }
76.
77.     public function setTglLahirAttribute($value)
78.     {
79.         /** @var DateTime $value */
80.         $this->attributes['tgl_lahir'] = $value-
>format('Y-m-d');
81.     }
82.
83.     public function kelurahan()
84.     {
85.         return $this-
>belongsTo(Kelurahan::class, 'id_kelurahan', 'id'
);
86.     }
87.
88.     public function pendaftarans()
89.     {

```

```

90.         return $this-
>hasMany(Pendaftaran::class, 'id_pendaftaran', 'id');
91.     }
92. }

```

5.2.2.6. Model Pilihan

Tabel 5.2-10 Model Pilihan Sekolah

```

1. <?php
2.
3. namespace App\Core\Domain\Model;
4.
5. use Illuminate\Database\Eloquent\Model;
6.
7. /**
8.  * @property-read int $id
9.  * @property int $id_pendaftaran
10. * @property int $id_sekolah
11. * @property int $indeks_pilihan
12. */
13. class Pilihan extends Model
14. {
15.     protected $table = 'pilihan';
16.     public $timestamps = false;
17.
18.     protected $fillable = [
19.         'id_pendaftaran',
20.         'indeks_pilihan',
21.         'id_sekolah'
22.     ];
23.
24.     public function sekolah()
25.     {
26.         return $this-
>belongsTo(Sekolah::class, 'id_sekolah', 'id');
27.     }
28. }

```


5.2.2.7. Model Prestasi

Tabel 5.2-11 Model Prestasi

```
1. <?php
2.
3. namespace App\Core\Domain\Model;
4.
5. use Illuminate\Database\Eloquent\Model;
6.
7. class Prestasi extends Model
8. {
9.     // Jenis Prestasi
10.    public const AKADEMIK_NONAKADEMIK = 'akademik
    /nonakademik';
11.    public const HAFIDZ = 'hafidz';
12.
13.    // Kamus Singkatan
14.    // AKADEMIK/NONAKADEMIK = AKADNON
15.    // PERORANGAN = PERORG
16.    // BEREGU = REGU
17.    // INTERNASIONAL = INTER
18.    // NASIONAL = NAS
19.    // PROVINSI = PROV
20.    // KABUPATEN = KAB
21.    // JUARA 1 = 1
22.    // JUARA 2 = 2
23.    // JUARA 3 = 3
24.
25.    // HAFIZH = HAFIZH
26.    // HAFAL 1 JUZ = 1_JUZ
27.    // HAFAL 2 JUZ = 2_JUZ
28.    // HAFAL 3 JUZ = 3_JUZ
29.    // HAFAL 4 JUZ = 4_JUZ
30.    // HAFAL > 4 JUZ = 5_JUZ
31.
32.    // Identitas Bobot
33.    public const AKADNON_PERORG_INTER_1 = 'API1';
34.    public const AKADNON_PERORG_INTER_2 = 'API2';
```

```

35.     public const AKADNON_PERORG_INTER_3 = 'API3';
36.     public const AKADNON_PERORG_NAS_1 = 'APN1';
37.     public const AKADNON_PERORG_NAS_2 = 'APN2';
38.     public const AKADNON_PERORG_NAS_3 = 'APN3';
39.     public const AKADNON_PERORG_PROV_1 = 'APP1';

40.     public const AKADNON_PERORG_PROV_2 = 'APP2';
41.     public const AKADNON_PERORG_PROV_3 = 'APP3';

42.     public const AKADNON_PERORG_KAB_1 = 'APK1';
43.     public const AKADNON_PERORG_KAB_2 = 'APK2';
44.     public const AKADNON_PERORG_KAB_3 = 'APK3';
45.
46.     public const AKADNON_REGU_INTER_1 = 'ABI1';
47.     public const AKADNON_REGU_INTER_2 = 'ABI2';
48.     public const AKADNON_REGU_INTER_3 = 'ABI3';
49.     public const AKADNON_REGU_NAS_1 = 'ABN1';
50.     public const AKADNON_REGU_NAS_2 = 'ABN2';
51.     public const AKADNON_REGU_NAS_3 = 'ABN3';
52.     public const AKADNON_REGU_PROV_1 = 'ABP1';
53.     public const AKADNON_REGU_PROV_2 = 'ABP2';
54.     public const AKADNON_REGU_PROV_3 = 'ABP3';
55.     public const AKADNON_REGU_KAB_1 = 'ABK1';
56.     public const AKADNON_REGU_KAB_2 = 'ABK2';
57.     public const AKADNON_REGU_KAB_3 = 'ABK3';
58.
59.     public const HAFIZH_1_JUZ = 'HHH1';
60.     public const HAFIZH_2_JUZ = 'HHH2';
61.     public const HAFIZH_3_JUZ = 'HHH3';
62.     public const HAFIZH_4_JUZ = 'HHH4';
63.     public const HAFIZH_5_JUZ = 'HHH5';
64.
65.
66.     // Robot
67.     public const BOBOT_AKADNON_PERORG_INTER_1 = 1
    00;
68.     public const BOBOT_AKADNON_PERORG_INTER_2 = 9
    0;
69.     public const BOBOT_AKADNON_PERORG_INTER_3 = 8
    0;

```

```

70.   public const BOBOT_AKADNON_PERORG_NAS_1 = 50;
71.   public const BOBOT_AKADNON_PERORG_NAS_2 = 45;
72.   public const BOBOT_AKADNON_PERORG_NAS_3 = 40;
73.   public const BOBOT_AKADNON_PERORG_PROV_1 = 25
;
74.   public const BOBOT_AKADNON_PERORG_PROV_2 = 23
;
75.   public const BOBOT_AKADNON_PERORG_PROV_3 = 20
;
76.   public const BOBOT_AKADNON_PERORG_KAB_1 = 13;
77.   public const BOBOT_AKADNON_PERORG_KAB_2 = 11;
78.   public const BOBOT_AKADNON_PERORG_KAB_3 = 9;

79.
80.   public const BOBOT_AKADNON_REGU_INTER_1 = 75;
81.   public const BOBOT_AKADNON_REGU_INTER_2 = 67;
82.   public const BOBOT_AKADNON_REGU_INTER_3 = 60;

83.   public const BOBOT_AKADNON_REGU_NAS_1 = 37;
84.   public const BOBOT_AKADNON_REGU_NAS_2 = 33;
85.   public const BOBOT_AKADNON_REGU_NAS_3 = 30;
86.   public const BOBOT_AKADNON_REGU_PROV_1 = 19;

87.   public const BOBOT_AKADNON_REGU_PROV_2 = 17;

88.   public const BOBOT_AKADNON_REGU_PROV_3 = 15;

89.   public const BOBOT_AKADNON_REGU_KAB_1 = 9;
90.   public const BOBOT_AKADNON_REGU_KAB_2 = 8;
91.   public const BOBOT_AKADNON_REGU_KAB_3 = 6;
92.
93.   public const BOBOT_HAFIZH_1_JUZ = 15;
94.   public const BOBOT_HAFIZH_2_JUZ = 25;
95.   public const BOBOT_HAFIZH_3_JUZ = 35;
96.   public const BOBOT_HAFIZH_4_JUZ = 40;

```

```

97.     public const BOBOT_HAFIZH_5_JUZ = 50;
98.
99.
100.    protected $table = 'prestasi';
101.    public $timestamps = false;
102.
103.    protected $fillable = [
104.        'id_pendaftaran_prestasi',
105.        'urutan_prestasi',
106.        'jenis_prestasi',
107.        'identitas_bobot',
108.        'bobot',
109.        'is_valid'
110.    ];
111.    }

```

5.2.2.8. Model Sekolah

Tabel 5.2-12 Model Sekolah

```

1.  <?php
2.
3.  namespace App\Core\Domain\Model;
4.
5.  use Illuminate\Database\Eloquent\Model;
6.
7.  /**
8.   * @property-read int $id
9.   * @property string $nama_sekolah
10.  * @property int $zona
11.  * @property float $latitude
12.  * @property float $longitude
13.  */
14.  class Sekolah extends Model
15.  {
16.      protected $table = 'sekolah';
17.      public $timestamps = false;
18.
19.      protected $fillable = [
20.          'nama_sekolah',

```



```

21.     'zona',
22.     'latitude',
23.     'longitude'
24. ];
25.
26.     // public function zona()
27.     // {
28.     //     $this-
>belongsTo(Zona::class, 'zona', 'id');
29.     // }
30. }

```

5.2.2.9. Model Zona

Tabel 5.2-13 Model Zona Sekolah

```

1. <?php
2.
3. namespace App\Core\Domain\Model;
4.
5. use Illuminate\Database\Eloquent\Model;
6.
7. /**
8.  * Model Zona. Kemungkinan tidak terpakai, hanya
9.  * untuk mengisi tabel rekomendasi
10.  *
11.  * @property-read int $id
12.  * @property int $nomor_zona
13.  */
14. class Zona extends Model
15. {
16.     protected $table = 'zona';
17.     public $timestamps = false;
18.
19.     protected $fillable = [
20.         'nomor_zona'
21.     ];
22.     // public function kelurahans()
23.     // {

```

```
24.     // $this-
      >belongsToMany(Kelurahan::class, 'kelurahan_zona'
      , 'id_zona', 'id_kelurahan', 'id');
25.     // }
26. }
```

[Halaman ini sengaja dikosongkan]

BAB 6

KESIMPULAN DAN SARAN

6.1. Kesimpulan

Kesimpulan yang didapat berdasarkan pengalaman saat melakukan pembuatan Aplikasi Penerimaan Peserta Didik Baru (PPDB) Kabupaten Bojonegoro, Jawa Timur tahun ajar 2020/2021 adalah:

- Sistem aplikasi yang dibangun oleh penulis dan tim pengembang lainnya telah sesuai dengan prosedur standar operasional dari Dinas Pendidikan Kabupaten Bojonegoro, Jawa Timur dan beroperasi dengan lancar.
- Aplikasi Penerimaan Peserta Didik Baru (PPDB) Kabupaten Bojonegoro, Jawa Timur tahun ajar 2020/2021 dapat digunakan oleh masyarakat sekitar Kabupaten Bojonegoro sebagai aplikasi pendaftaran ke jenjang SMP bagi siswa SD dan sebagai aplikasi manajemen pendaftaran untuk pihak sekolah jenjang menengah pertama yang berpartisipasi kepada program PPDB.

6.2. Saran

Pengerjaan Aplikasi Penerimaan Peserta Didik Baru (PPDB) Kabupaten Bojonegoro, Jawa Timur tahun ajar 2020/2021 jauh dari kata sempurna, oleh karena itu penulis akan menyampaikan beberapa saran untuk pengembangan aplikasi serupa ke depannya, yaitu:

- Membangun aplikasi dengan arsitektur yang lebih jelas, sehingga ketika akan digunakan kembali untuk program selanjutnya menjadi lebih mudah untuk dirawat (*maintenance*), memudahkan pembacaan kode program, dan meringankan beban yang tidak perlu pada *server*.
- Mendesain *database* dengan matang, sehingga jika terjadi penambahan *table* baru menjadi lebih mudah untuk dirawat (*maintenance*). Selain itu mendesain *database* dengan baik akan meningkatkan performa akses baca dan tulis terlebih saat pengguna mengakses ke aplikasi secara massal.
- Menggunakan *framework* yang lebih populer dengan dokumentasi yang lebih jelas dan detail.

Beberapa *framework* yang dapat digunakan adalah *Laravel*, *Django*, ataupun *Phalcon*.

[Halaman ini sengaja dikosongkan]

BIODATA PENULIS



Akwila Feliciano Pradiptatmaka lahir di Jakarta pada tanggal 21 April 1999. Penulis merupakan mahasiswa yang menempuh studi di Departemen Teknik Informatika, Fakultas Teknologi Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember, Surabaya. Biasa menggunakan *stack* Javascript, Python, dan PHP untuk membangun aplikasi berbasis *website*.



Setya Wibawa lahir di Surabaya pada 12 April 1999. Penulis merupakan mahasiswa yang menempuh studi di Departemen Teknik Informatika Institut Teknologi Sepuluh Nopember (ITS)