

Implementasi Model View Controller Untuk Proses Pencatatan dan Monitoring Pelanggaran Siswa Pada SMK Satria

Rakasiwi Akbar^{1*}, Mochammad Abdul Azis²

^{1,2}Universitas Bina Sarana Informatika
Jl. Kramat Raya No 98, Senen, Jakarta Pusat, Indonesia

e-mail korespondensi: rakasiwiakbar2@gmail.com

Informasi Artikel: Submit: 19-09-2023 | Revisi : 30-09-2023 | Terima : 27-10-2023

Abstrak - SMK SATRIA merupakan sekolah menengah kejuruan yang memiliki 5 jurusan yang berlokasi di Srengseng. Pada sekolah ini belum mempunyai sistem informasi berbasis *website* yang dapat mempermudah dalam melakukan proses mencatat pelanggaran yang dilakukan siswa. Maka dari itu dibutuhkan sebuah sistem berbasis *website* yang dapat digunakan untuk mencatat pelanggaran siswa agar prosesnya lebih cepat, aman dan terkomputerisasi. Sistem ini dibangun dengan pendekatan berorientasi objek. Metode yang digunakan yaitu (MVC) *Model, view, Controller*. Metode ini menggunakan bahasa PHP dalam pengkodean, MySQL sebagai *database* dan CodeIgniter 3 sebagai *framework*. Pada proses perancangan sistem ini melibatkan bagian-bagian dari UML (*Unified Modeling Language*) seperti *use case diagram, activity diagram, sequence diagram, dan deployment diagram*. Sistem informasi ini dirancang dengan beberapa halaman, seperti halaman login, dashboard, tipe pelanggaran, list pelanggaran, data siswa, class, guru, pengguna dan laporan. Hasilnya berupa sebuah sistem yang dapat membantu pihak sekolah untuk *memonitoring* pelanggaran yang dilakukan siswa dan mengurangi resiko kesalahan dalam melakukan pencatatan pelanggaran siswa.

Kata Kunci : Pelanggaran Siswa, Perancangan, MVC, UML

Abstracts - SMK SATRIA is a vocational high school that has 5 majors located in Srengseng. This school does not yet have a website-based information system that can facilitate the process of recording student violations. Therefore a website-based system is needed that can be used to record violations. students so that the process is faster, safer and computerized. This system is built with an object-oriented approach. The method used is (MVC) *Model, view, Controller*. This method uses the PHP language for coding, MySQL as the database and CodeIgniter 3 as the framework. The process of designing this system involves parts of the UML (*Unified Modeling Language*) such as *use case diagrams, activity diagrams, sequence diagrams, and deployment diagrams*. This information system is designed with several pages, such as *login page, dashboard, types of violations, list of violations, student data, classes, teachers, users and reports*. The result is a system that can help the school to monitor violations committed by students and reduce the risk of errors in recording student violations.

Keywords : Student Violations, Designing, MVC, UML

1. Pendahuluan

Perkembangan teknologi tidak dapat dihindari di era digital sehingga dapat dimanfaatkan secara umum. Pemanfaatan teknologi ini harus mampu memudahkan operasional dan aktivitas sehari-hari [1]. Sekolah merupakan suatu lembaga pendidikan yang memiliki misi untuk membentuk diri dan kepribadian peserta didik agar memiliki kepribadian yang baik dan kedisiplinan yang tinggi [2]. Salah satu cara agar dapat melatih dan meningkatkan kedisiplinan siswa adalah dengan menerapkan tata tertib sekolah untuk mengatur sikap, kedisiplinan, tingkah laku dan tanggung jawab.

Pada pelaksanaannya, setiap harinya ada saja siswa yang melakukan pelanggaran peraturan sekolah. Namun dalam penerapannya proses pencatatan pelanggaran masih belum efektif dan efisien. Setiap pelanggaran yang dilakukan oleh siswa akan dicatat dalam buku pelanggaran siswa yang membuat proses pencatatan menjadi lama. Untuk melihat pelanggaran dan rekap data siswa juga membutuhkan waktu yang lama dikarenakan harus melihat dan mencari secara manual pada buku pelanggaran [3].

Dalam perkembangan teknologi yang semakin pesat, *website* juga dapat digunakan untuk membantu dalam seluruh aspek kegiatan. *website* dapat dijelaskan sebagai kumpulan halaman yang berisi berupa text, gambar, video dan animasi lainnya dengan koneksi internet [2].



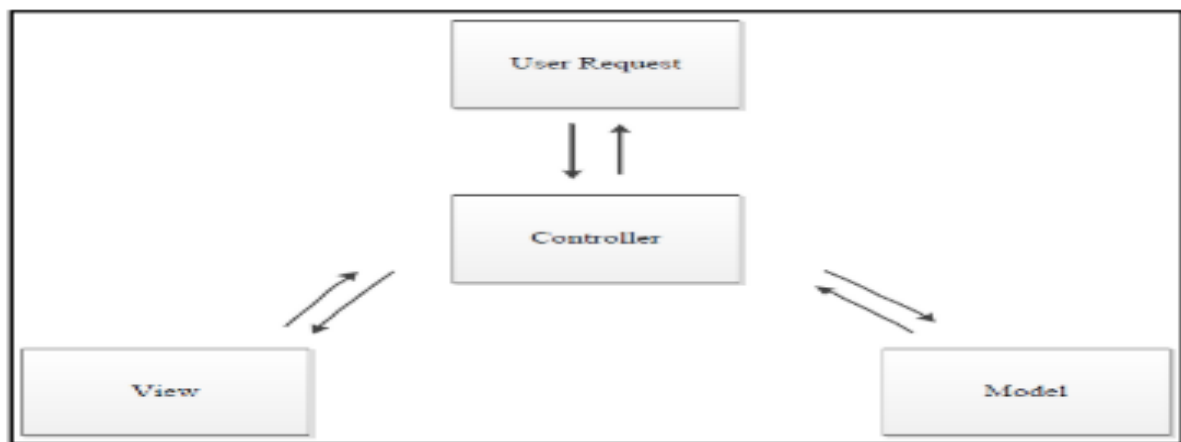
Konsep MVC merupakan salah satu arsitektur dalam membuat sebuah aplikasi berbasis web. konsep ini menggunakan 3 komponen yaitu *Model*, *View* dan *Controller*, dengan tujuan untuk mempermudah dalam mengembangkan suatu program [4].

Berdasarkan masalah tersebut, maka diperlukan sebuah program yang dapat melakukan proses pencatatan pelanggaran secara efektif, serta dapat melakukan rekap data siswa yang melakukan pelanggaran. Dengan menggunakan konsep MVC diharapkan dapat mempermudah dalam pembuatan program pencatatan pelanggaran dan rekap data siswa agar dapat tersimpan secara terkomputerisasi.

2. Metode Penelitian

Model pengembangan perangkat lunak yang dipergunakan oleh penyusun dalam sistem ini mempergunakan metode *Model View controller (MVC)* untuk membantu memecahkan permasalahan dalam sistem pencatatan dan monitoring pelanggaran siswa [5].

Model View Controller atau MVC merupakan salah satu dari pola *design* yang pada umumnya dipergunakan untuk mengembangkan *interface* pengguna yang membagi logika program kedalam tiga bagian yang saling berhungan. Hal ini dilakukan untuk memisahkan cara informasi disajikan secara internal dan diterima dari pengguna. Berikut adalah gambaran alur tahapan MVC.



Sumber: [4]

Gambar 1. Alur Tahapan MVC

Berikut adalah penjelasan dari tahapan-tahapan dari MCV:

1. Model

Model biasanya berkaitan dengan informasi yang berada pada *database*. *Model* juga menjelaskan struktur data, yang dapat berupa *database* atau data lain seperti file teks, file XML, maupun *webservice*. Secara umum *model* memiliki fungsi untuk mengambil, memperbarui, dan menghapus informasi dari sebuah web. Aplikasi web biasanya menggunakan *database* untuk menyimpan data, sehingga *model* biasanya berisi perintah-perintah *query SQL* [6].

2. View

View merupakan bagian yang memiliki fungsi sebagai penerima dan menampilkan hasil dari *model* dan *controller* kepada pengguna. Pada bagian ini biasanya suatu aplikasi web berisi file HTML yang diatur pada *controller* [6].

3. Controller

Controller adalah bagian yang menentukan untuk menghubungkan antara bagian *model* dan *view*. Pada *controller*, ada kelas dan fungsi yang bertujuan untuk memproses permintaan dari bagian *view* ke bagian *model*. *Controller* memiliki tugas untuk mempersiapkan macam-macam dari *variabel* untuk ditampilkan kepada *user*, memanggil *model* untuk mengakses *database*, melakukan validasi [6].

3. Hasil dan Pembahasan

Pada saat merancang sistem pencatatan dan monitoring pelanggaran siswa, peneliti melakukan analisis terhadap kebutuhan yang muncul dalam mengembangkan sistem pencatatan dan monitoring pelanggaran siswa pada SMK SATRIA. Di bawah ini adalah spesifikasi sistem pencatatan dan monitoring pelanggaran siswa:

1. Kebutuhan Sistem
 - a. Admin

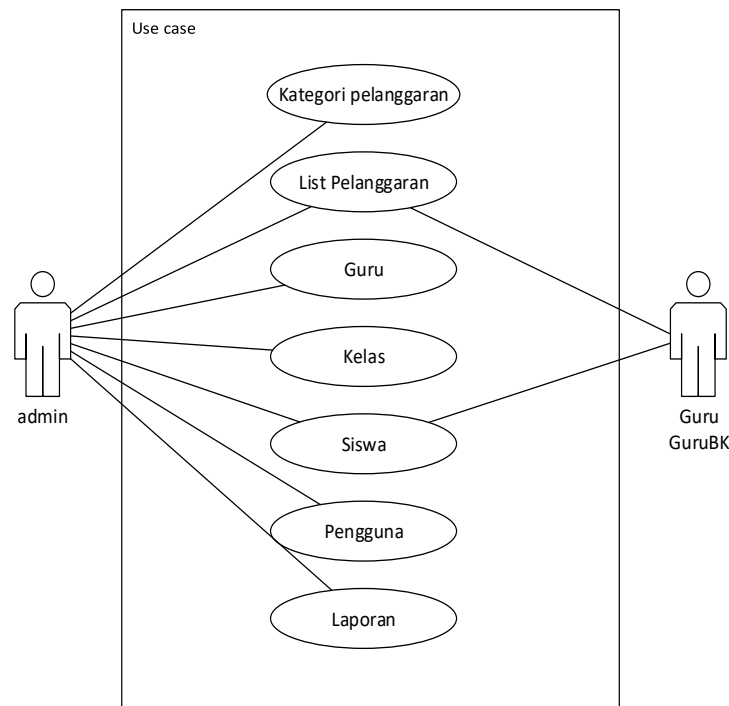
- 1) Admin dapat melakukan *login*.
- 2) Admin dapat mengelolah data siswa.
- 3) Admin dapat mengelolah data kategori pelanggaran.
- 4) Admin dapat mengelolah data guru.
- 5) Admin dapat mengelolah data kelas.
- 6) Admin dapat melakukan input data pelanggaran.
- 7) Admin dapat mengelolah data laporan pelanggaran.

b. Guru

- 1) Guru dapat menginput data pelanggaran.
- 2) Guru dapat melihat data siswa.
- 3) Guru dapat melihat data pelanggaran.

2. *Use Case Diagram*

Use case diagram dapat menjelaskan hubungan antara aktor dengan sistem dengan sistem. Secara kasar *use case* digunakan untuk mengetahui fungsi apa saja yang tersedia pada sistem informasi dan siapa saja yang boleh menggunakannya [7]. Merancang sebuah *use case* diagram bertujuan untuk menjelaskan interaksi yang terjadi antara sistem dengan aktor. *Use case* diagram juga dapat dipergunakan untuk mengetahui tipe interaksi antara pengguna sistem dengan sistem itu sendiri. Berikut gambar 2 adalah *desain use case diagram* sistem Pencatata.

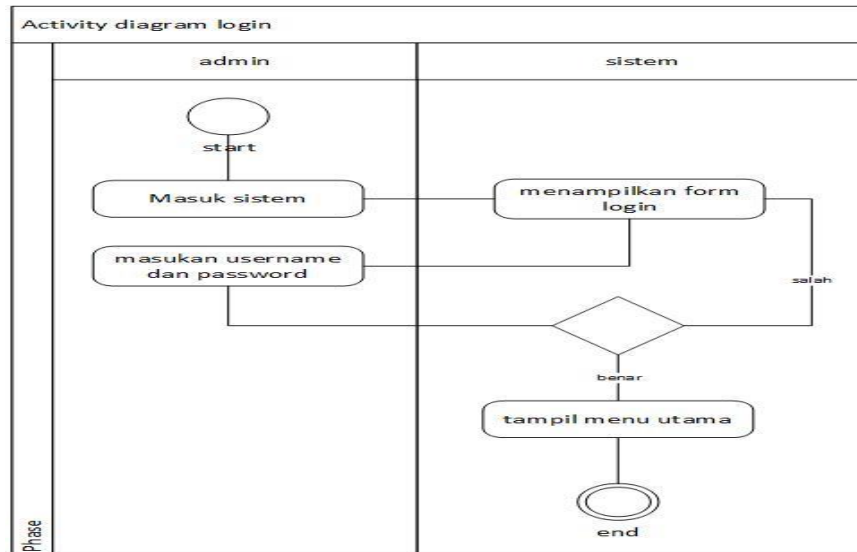


Gambar 2. *Use case diagram*

Gambar 2 merupakan *use case digram* sistem pencatatan dan monitoring pelanggaran siswa. Dimana untuk admin dapat mengakses kategori pelanggaran, *list* pelanggaran, data guru, data kelas, data siswa, pengguna dan laporan sedangkan untuk guru hanya dapat mengakses *list* pelanggaran dan data siswa.

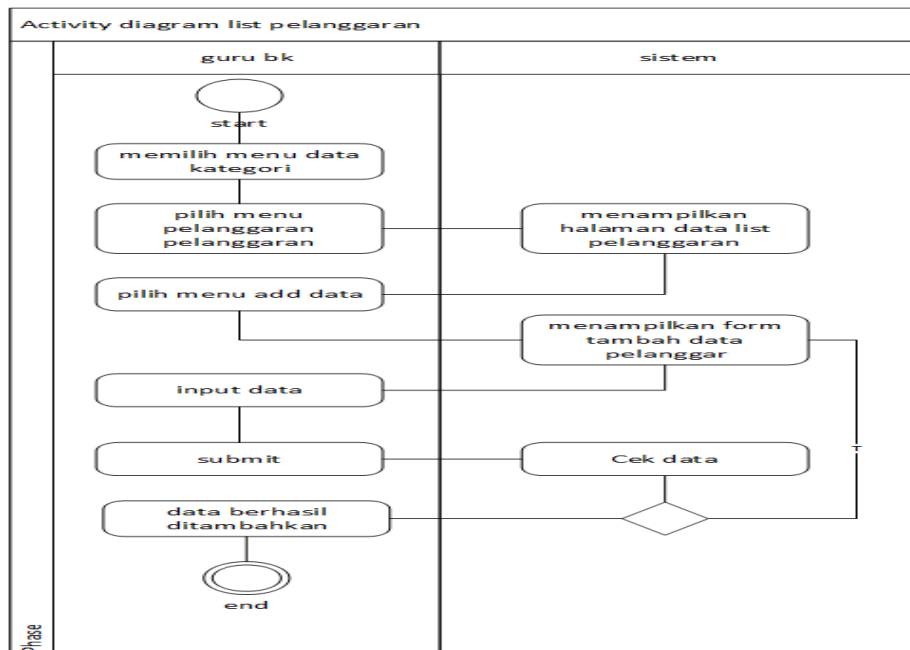
3. *Activity Diagram*

Diagram yang menggambarkan atau memodelkan berbagai aliran aktivitas dalam sistem pengembangan dan bagaimana setiap aliran dimulai dengan kemungkinan keputusan dan bagaimana setiap aktivitas berakhir.[8] Memberikan sebuah gambaran sebuah proses pada sistem yang dibuat agar lebih mudah memahami yang terdapat dalam sistem secara mendalam dan mengetahui aktivitas aktor berdasarkan *use case* yang sudah dibuat sebelumnya.



Gambar 3. Activity Diagram Login

Pada gambar 3 merupakan *activity diagram login* untuk admin dan guru. Dimanana aktivitas ini mulai dengan *user* mengakses sistem, kemudian sistem akan menampilkan form yang berisikan *username* dan *password* kemudian form tersebut akan diisi oleh *user*. Setelah diisi sistem akan mengecek apakah *username* dan *password* sudah benar atau tidak jika sudah sesuai maka sistem akan menampilkan halaman utama jika tidak akan kembali ke form.

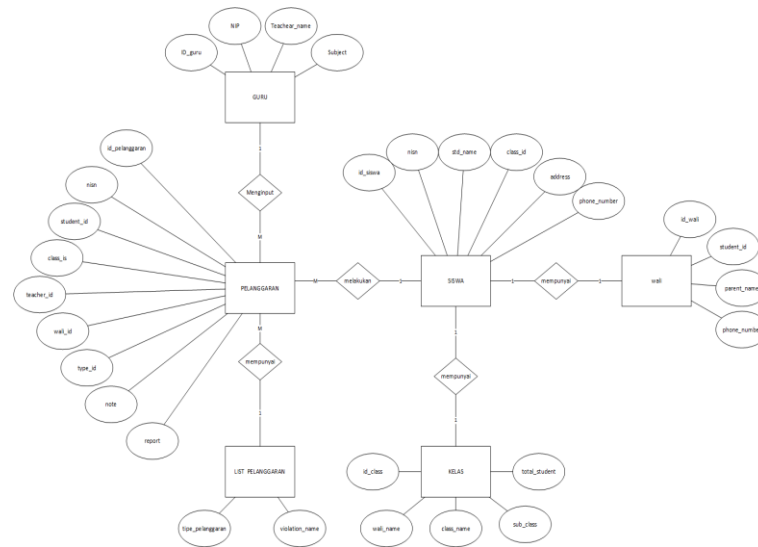


Gambar 4. Activity Diagram List Pelanggaran

Pada gambar 4 adalah *activity diagram list pelanggaran* untuk melakukan pencatatan pelanggaran siswa. Dimulai dari memilih menu data kategori kemudian pilih pelanggaran, kemudian sistem akan menampilkan halaman data *list* pelanggaran. Selanjutnya pilih menu add untuk mencatat pelanggaran, kemudian sistem akan menampilkan form. Kemudian input data siswa yang melanggar dan jenis pelanggarannya selanjutnya submit. Jika semua data sudah disii maka akan ada tulisan data berhasil ditambahkan, sedangkan jika tidak akan kembali ke form tambah data.

4. *Entity Relationship Diagram (ERD)*

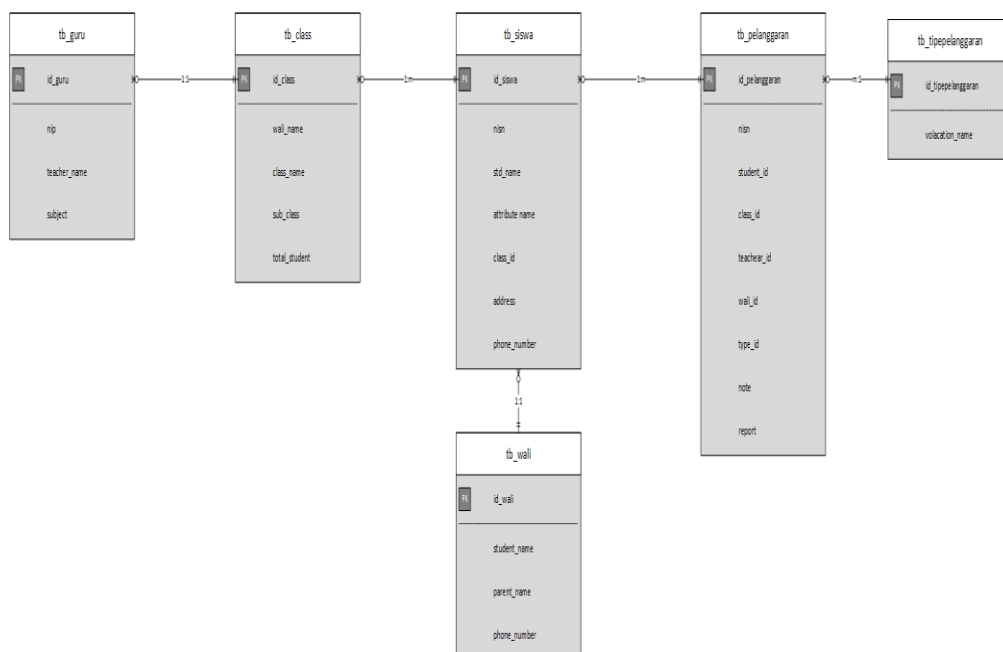
Menurut Marlinda dalam Tabrani menjelaskan bahwa, “*Model Entity Relationship* merupakan suatu *model* untuk menjelaskan hubungan antar data dalam *basis data* berdasarkan suatu persepsi bahwa *realworld* terdiri dari *object-object* dasar yang mempunyai hubungan atau relasi antar *object- object* tersebut” [9].*Entity Relationship Diagram* menjelaskan hubungan basis data yang digunakan untuk model basis data relasional. Gambar 5 dibawah ini merupakan gambaran dari *entity relationship diagram* sistem pencatatan dan monitoring pelanggaran siswa.



Gambar 5. *Entity Relationship Diagram*

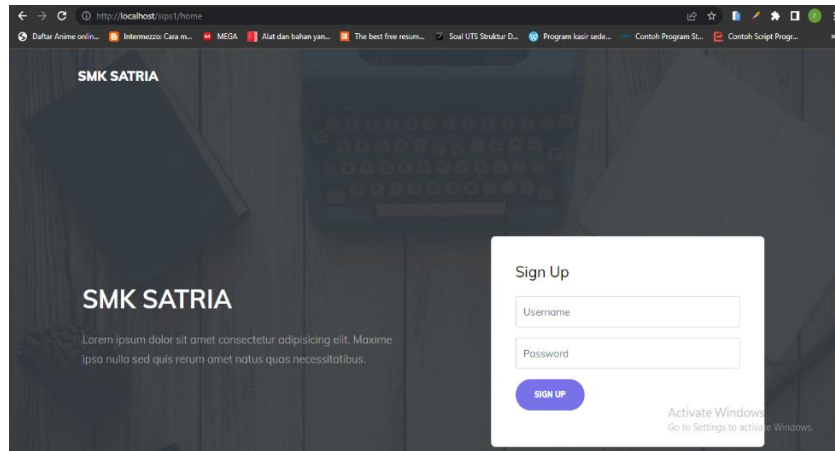
5. *Logical Record Structure (LRS)*

Menurut Dhanta dalam jurnal [10] menjelaskan “*LRS (Logical Record Structure)* adalah representasi dari struktur record- record pada tabel-tabel yang terbentuk dari hasil antara himpunan entitas. Menentukan kardinalitas, jumlah tabel dan Foreign Key (FK)”. Sedangkan menurut Friyadie dalam [11] menjelaskan “*LRS* merupakan hasil dari permodelan *Entity RelationalShip* (ER) beserta atributnya sehingga bisa terlihat hubungan-hubungannya antar entitas”. Gambar 6 merupakan gambar dari *logical record structure* dari sistem pencatatan dan monitoring pelanggaran siswa.



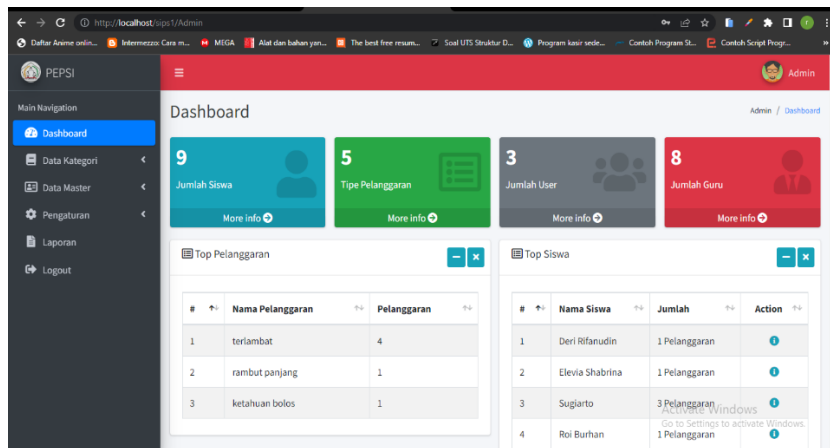
Gambar 6. *Design Logical Record Structure*

6. *User Interface*
Halaman Login



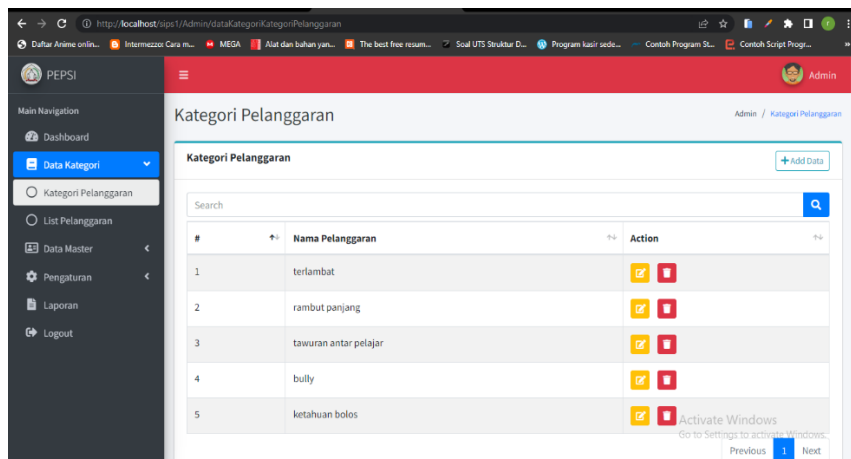
Gambar 7. Tampilan Halaman *Login*

Halaman Dashboard admin



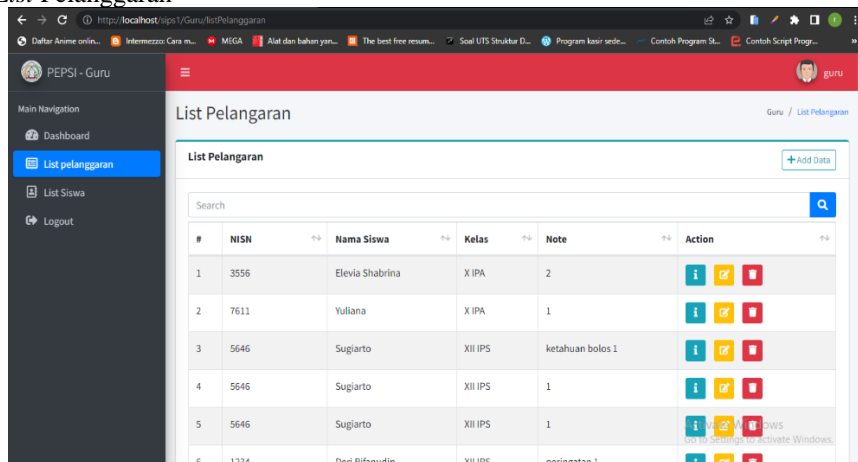
Gambar 8. Tampilan Halaman Dashboard admin

Halaman Kategori Pelanggaran Admin



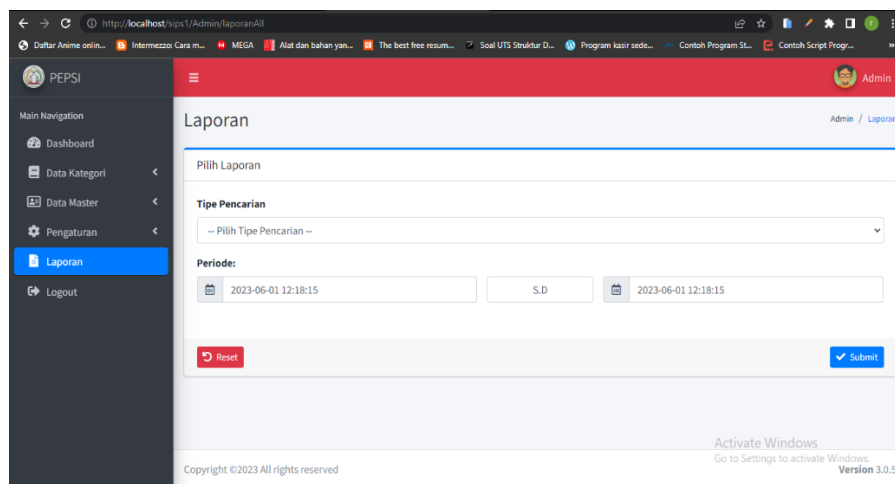
Gambar 9. Tampilan Halaman Kategori Pelanggaran Admin

Halaman *List Pelanggaran*



Gambar 10. Tampilan Halaman *List Pelanggaran*

Halaman Laporan Pelanggaran



Gambar 11. Tampilan Halaman Laporan

4. Kesimpulan

Berdasarkan hasil perancangan program pencatatan dan monitoring pelanggaran siswa ini, maka dapat disimpulkan dengan adanya sistem ini dapat menaikkan tingkat efektifitas dalam proses pencatatan dan monitoring pelanggaran yang telah dilakukan siswa. Dengan adanya sistem pencatatan dan monitoring ini dapat membantu mempercepat dalam melakukan pencatatan siswa yang melakukan pelanggaran. Dengan menggunakan sistem ini dapat mempermudah dalam melakukan rekap data siswa yang melakukan pelanggaran. Dan dengan adanya sistem ini dapat membantu untuk mengetahui pelanggaran apa saja yang sering dilanggar oleh siswa.

Referensi

- [1] L. Setiawati, E. Siti Aulia, R. Cynthia Johan, A. Hadiapurwa, and Ardiansah, "Studi Analisis Website Sistem Informasi Kearsipan Dinamis (SIKD)," *J. Pengemb. Kearsipan*, vol. 15, no. 1, pp. 84–97, 2022.
- [2] S. E. T. Buana, L. H. Atrinawati, and M. G. L. Putra, "Penerapan Metode Agile Untuk Membangun Sistem Informasi Monitoring Santri Pondok Modern Asy-Syifa Balikpapan," *Seminar.Iaii.or.Id*, pp. 183–190, 2021, [Online]. Available: <http://www.seminar.iaii.or.id/index.php/SISFOTEK/article/view/280>
- [3] O. Mabuka, "Tata Tertib Sekolah Berperan Sebagai Pengendali Perilaku Siswa di SD Inpres Raja Kecamatan Morotai Selatan Barat," *J. Ilm. Wahana Pendidik.*, vol. 7, no. 2, pp. 360–372, 2021, doi: 10.5281/zenodo.4724351.
- [4] Y. Anggraini, D. Pasha, D. Damayanti, and A. Setiawan, "Sistem Informasi Penjualan Sepeda Berbasis

- Web Menggunakan Framework Codeigniter,” *J. Teknol. dan Sist. Inf.*, vol. 1, no. 2, pp. 64–70, 2020, doi: 10.33365/jtsi.v1i2.236.
- [5] N. Hidayati, “Pengembangan Sistem Informasi Pembelian Obat Secara Kredit Pada Apotik Dengan Menggunakan Model View Controller (MVC),” *J. Sains Komput. Inform. (J-SAKTI)*, vol. 4, no. 2, pp. 457–471, 2020.
- [6] S. Irsyad and A. S. Sitio, “Penerapan Konsep Mvc Pada Sistem Penjualan Online Dengan Sistem Keamanan Menggunakan Algoritma Rijndael,” *INFORMATIKA*, vol. 11, no. 2, p. 44, 2019, doi: 10.36723/juri.v11i2.168.
- [7] R. Hormati, S. Yusuf, and M. Abdurahman, “Sistem informasi Data Poin Pelanggaran Siswa Menggunakan Metode Prototyping Berbasis Web Pada SMA Negeri 10 Kota,” *J. Ilm. Ilk. - Ilmu Komput. Inform.*, vol. 4, no. 2, pp. 93–103, 2021, doi: 10.47324/ilkominfo.v4i2.128.
- [8] A. Noviantoro, A. B. Silviana, R. R. Fitriani, and H. P. Permatasari, “Rancangan Dan Implementasi Aplikasi Sewa Lapangan Badminton Wilayah Depok Berbasis Web,” *J. Tek. dan Sci.*, vol. 1, no. 2, pp. 88–103, 2022, doi: 10.56127/jts.v1i2.108.
- [9] M. Tabrani and I. Rezqy Aghniya, “Implementasi Metode Waterfall Pada Program Simpan Pinjam Koperasi Subur Jaya Mandiri Subang,” *J. Interkom J. Publ. Ilm. Bid. Teknol. Inf. dan Komun.*, vol. 14, no. 1, pp. 44–53, 2020, doi: 10.35969/interkom.v14i1.65.
- [10] T. Hartati, N. Anastia, and R. Widyastuti, “Penerapan SDLC Model Waterfall pada Rancang Bangun SI-PKP Direktorat Jenderal ILMATE Kementerian Perindustrian Jakarta,” *Remik*, vol. 6, no. 1, pp. 9–15, 2021, doi: 10.33395/remik.v6i1.11127.
- [11] I. Rusdi, A. S. Mulyani, and I. Herlina, “Rancang Bangun Sistem Informasi Pembelian Pada Cv. Cimanggis Jaya Depok,” *J. AKRAB JUARA*, vol. 5, no. 2, pp. 180–197, 2020, [Online]. Available: <http://akrabjuara.com/index.php/akrabjuara/article/view/1012>