

Perancangan Dan Pembuatan Halaman Admin Sistem Informasi Surat Tugas di Fakultas Teknik Universitas Diponegoro Menggunakan Kerangka Kerja Laravel

M. Gesit Alifandi*, Agung Budi Prasetijo, Adnan Fauzi

Departemen Teknik Komputer, Fakultas Teknik, Universitas Diponegoro
Jl. Prof. Sudarto No.13, Tembalang, Kec. Tembalang, Kota Semarang, Jawa Tengah 50275
fandi7418@students.undip.ac.id*, agungprasetijo@lecturer.undip.ac.id, adnan@ft.undip.ac.id

ABSTRACT

Pada era digital 4.0 ini, masih terdapat proses pembuatan dan perizinan surat tugas yang dilakukan oleh dosen dan staf secara manual. Proses pembuatan nomor surat, perizinan ke Ketua Departemen, Supervisor, dan Wakil Dekan masih dilakukan secara manual. Selain itu, belum adanya peranan seorang admin dalam mengelola data pengguna dan data surat dikarenakan sistem masih dikerjakan secara manual. Hal tersebut masih memiliki beberapa kekurangan seperti kurangnya efisiensi dalam pengerjaan. Oleh karena itu, dibutuhkan sebuah sistem informasi yang diharapkan dapat membantu proses perizinan surat tugas dan pengelolaan data pengguna maupun data surat. Dalam hal ini, Penulis membuat sistem informasi yang dikhususkan untuk seorang admin agar dapat mengelola data pengguna yang terlibat maupun data surat di dalam sistem informasi tersebut. Sistem Informasi Surat Tugas pada halaman admin ini dikembangkan menggunakan metode Waterfall karena memberi kemudahan dalam perancangan sistem informasi dengan pendekatan yang sistematis. Pengembangan Sistem Informasi Surat Tugas ini dirancang menggunakan kerangka kerja Laravel karena memiliki konsep MVC (Model, View, Controller) yang mempermudah developer dalam mengembangkan sebuah website. Sistem Informasi Surat Tugas yang telah dinyatakan selesai, diuji menggunakan metode black-box. Berdasarkan hasil pengujian black-box, sistem dapat berjalan dengan baik sesuai perancangan sistem yang telah dibuat.

Kata Kunci: Sistem Informasi, Laravel, Black-box, admin, waterfall

PENDAHULUAN

Fakultas Teknik Universitas Diponegoro merupakan salah satu fakultas yang ada di Universitas Diponegoro. Fakultas Teknik diawali dengan dibukanya Akademi Teknik pada tanggal 20 Oktober 1958 yang awalnya berdiri sejak 9 Januari 1957. Salah satu kegiatan yang ada di Fakultas Teknik adalah kegiatan administrasi diantaranya yaitu pendataan mahasiswa, pendataan dosen, pendataan program studi, dan lain sebagainya. Diantara banyaknya kegiatan administrasi tersebut, salah satunya adalah pembuatan surat tugas dosen dan staf saat akan melaksanakan kegiatan di luar kampus.

Pembuatan surat tugas dosen dan staf pada Fakultas Teknik Universitas Diponegoro masih menggunakan cara manual. Setiap dosen dan staf yang akan melakukan

kegiatan di luar kampus diharuskan untuk mencetak surat lalu meminta tanda tangan ke ketua departemen, supervisor, dan wakil dekan. Selain itu, belum adanya peranan seorang admin dalam mengelola data pengguna dan data surat dikarenakan sistem masih dikerjakan secara manual. Hal tersebut membuat proses perizinan surat kurang efektif dan efisien.

Perkembangan teknologi digital yang sangat pesat di era digital 4.0 memberikan banyak manfaat dalam kemajuan aspek kehidupan manusia. Hampir seluruh kegiatan manusia sekarang ini dijalankan secara online. Hal ini memberikan kemudahan dalam mengerjakan kegiatan sehari-hari. Hal tersebut dapat juga digunakan untuk mempermudah kegiatan administrasi surat tugas dosen dan staf di lingkungan Fakultas Teknik Universitas

Diponegoro. Oleh karena itu, Fakultas Teknik Universitas Diponegoro membutuhkan sebuah sistem informasi yang mempermudah proses perizinan surat tugas.

Berdasarkan permasalahan tersebut, maka bersama dengan rekan saya Julian Manuel bekerja sama untuk mengembangkan Sistem Informasi Surat Tugas yang diharapkan dapat memudahkan para dosen dan staf dalam membuat ataupun melakukan perizinan surat tugas sehingga proses yang dilakukan lebih sederhana, cepat, dan efisien. Dalam sistem informasi tersebut, dibutuhkan seorang admin yang berperan untuk mengelola data pengguna yang terlibat maupun data surat. Sistem informasi ini menggunakan metode perancangan waterfall. Metode waterfall merupakan pendekatan SDLC (Software Development Life Cycle) paling awal yang digunakan dalam pengembangan perangkat lunak. Sistem informasi ini dikembangkan menggunakan framework Laravel dengan konsep MVC (Model, View, Controller) (Sari, et al, 2020) yang mempermudah developer dalam mengembangkan sebuah website.

STUDI LITERATUR

Keberlangsungan penelitian ini tidak akan terlepas dari penelitian-penelitian terdahulu yang pernah dilakukan sebagai bahan pertimbangan dan perbandingan dalam membuat sebuah sistem informasi. Topik dari penelitian-penelitian tersebut akan dijadikan sebagai perbandingan pada penelitian ini sekaligus membantu dalam menyelesaikan masalah-masalah yang akan dihadapi selama penelitian berlangsung.

Penelitian yang dilakukan oleh Burhanuddin Damanik dengan judul “Perancangan Sistem Informasi SMP Negeri 1 Tuhemberua Kabupaten Nias Utara Menggunakan PHP Codeigniter” membahas tentang pemanfaatan *website* yang mempermudah dan memperluas profil sekolah dalam menyampaikan informasi secara online kepada siswa maupun masyarakat. Pengembangan sistem informasi pada penelitian ini menggunakan metode *waterfall* dengan bahasa pemrograman PHP dan menggunakan *framework* Codeigniter, serta menggunakan *tools* Visual Studio Code dan *database* MySQL (Damanik, 2021).

Penelitian selanjutnya yang dilakukan oleh Aginta Geniusia dan Febriliyan Samopa dengan judul “Perancangan Sistem Informasi Perjalanan Dinas Kantor Wilayah Direktorat Jenderal Perbendaharaan (SIPD-Kanwil DJPBN)” membahas tentang perancangan *website* untuk mempermudah bagian kepegawaian dan keuangan dalam mengelola perjalanan dinas. Pengembangan pada penelitian sistem informasi ini menggunakan PHP 5, *apache server*, *database* MySQL, dan berjalan dalam LAN (*Local Area Network*). (Geniusia & Samopa, 2013)

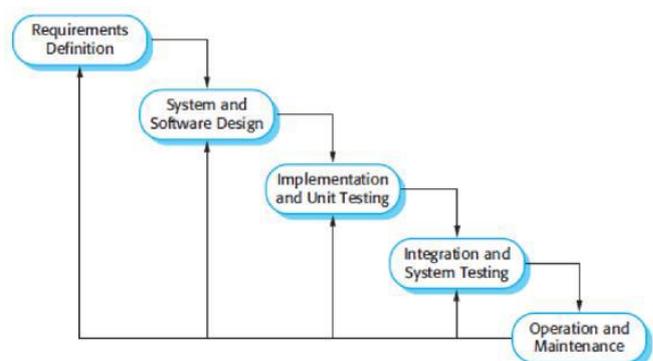
Penelitian selanjutnya dilakukan oleh Irza Haryo Prabowo dan Ari Kurniawan dengan judul “Rancang

Bangun Aplikasi Pengarsipan Surat dan Disposisi (APSD) Berbasis *Website* Menggunakan Framework Laravel” membahas tentang perancangan *website* untuk memudahkan administrasi Dinas Pertanian dan Ketahanan Pangan Kota Madiun dalam memproses penanganan, pengolahan, pengarsipan surat dan disposisi menjadi lebih efektif dan efisien. Pengembangan pada penelitian sistem informasi ini menggunakan *framework* Laravel dan menggunakan *database* MySQL. (Prabowo & Kurniawan, 2020)

Ketiga penelitian di atas dijadikan dasar penelitian dalam merancang dan mengembangkan aplikasi Sistem Informasi Surat Tugas. Ketiga penelitian tersebut mengambil tema sistem informasi dengan menggunakan MySQL sebagai *database*, Laravel dan CodeIgniter (Destiningrum & Adrian, 2017) sebagai *framework* (Wardana, 2010), dan metode *waterfall* sebagai metodologi penelitian. Penulis menggunakan MySQL sebagai *database* karena cukup efektif dalam pembuatan Sistem Informasi Surat Tugas. Penulis menggunakan *framework* Laravel karena memiliki beberapa fitur yang lebih lengkap dari *framework* lainnya (Banjarnahor & Hartomo, 2016) (Karyoto, 2014). Sedangkan dalam metodologi penelitian, Penulis menggunakan metode *waterfall* karena memiliki pendekatan yang sistematis, mulai dari tahap kebutuhan sistem, tahap analisis, desain, *coding*, *testing*, dan *maintenance*.

METODOLOGI

Perancangan aplikasi yang dilakukan pada penelitian ini menggunakan metode waterfall. Metode waterfall merupakan metode yang paling banyak digunakan dalam pengembangan aplikasi karena bersifat sistematis dan berurutan dalam pengembangan aplikasi. Terdapat 5 tahapan waterfall yang dapat dilihat pada Gambar 1.



Gambar 1. Waterfall

Adapun penjelasan tahapan-tahapan yang ada dalam metode *Waterfall* yaitu (Widiyanto, 2018):

a. Requirements *Definition* (Perencanaan Kebutuhan)

Tahapan dalam proses pengumpulan kebutuhan dilakukan secara intensif untuk menspesifikasikan kebutuhan perangkat lunak seperti apa yang dibutuhkan oleh pengguna.

b. System and Software Design (Desain Sistem)

Setelah mendapatkan informasi dari tahap *requirements definition*, informasi tersebut dianalisis kemudian diimplementasikan pada desain pengembangan. Pada tahap ini, pengembang menyiapkan kebutuhan sistem baik perangkat keras maupun perangkat lunak dengan membentuk sistem secara keseluruhan.

c. Implementation and Unit Testing (Implementasi)

Tahapan perancangan perangkat lunak mulai direalisasikan sebagai unit program. Pada tahap ini, pengembang melakukan verifikasi bahwa setiap unit sudah sesuai fungsi-fungsi yang diinginkan.

d. Integration and System Testing (Pengujian)

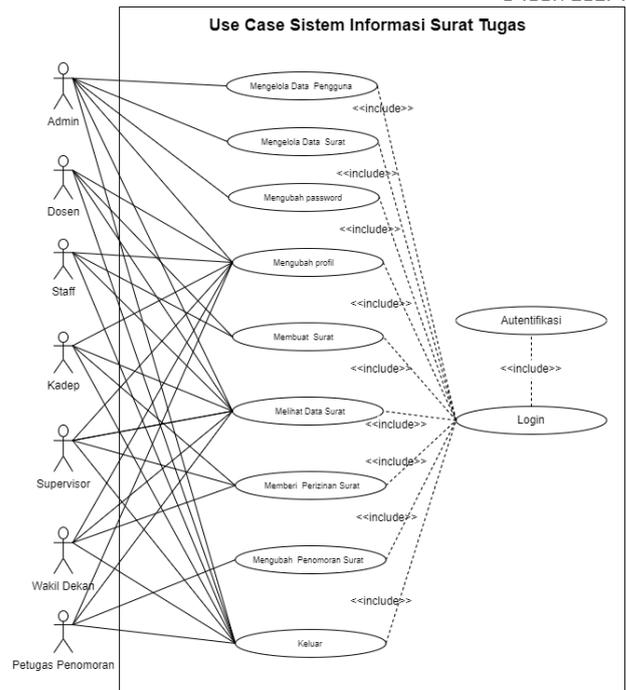
Tahapan penggabungan unit-unit program menjadi sebuah sistem perangkat lunak yang utuh dan dilakukan pengujian untuk mengetahui apakah sistem sudah sesuai atau masih ada kesalahan.

e. Operation and Maintenance (Pemeliharaan)

Perangkat lunak yang sudah jadi dijalankan oleh pengguna dan dilakukan pemeliharaan. Pemeliharaan memungkinkan pengembang melakukan perbaikan kesalahan yang tidak ditemukan pada tahap-tahap sebelumnya.

A. Diagram Case

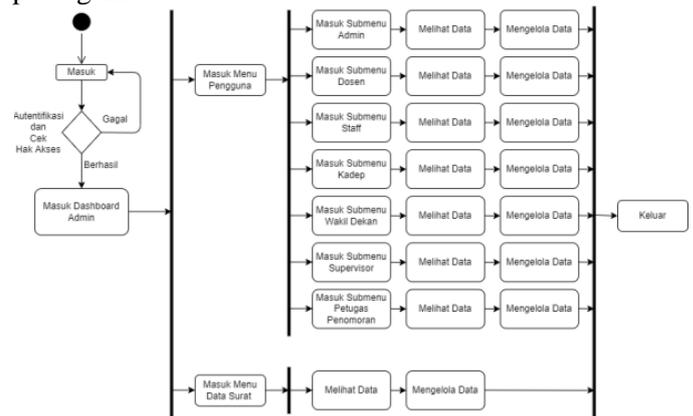
Gambar 2 menunjukkan diagram *use case* Sistem Informasi Surat Tugas pada Departemen Teknik Komputer Universitas Diponegoro



Gambar 2. Diagram Use Case Sistem

B. Diagram Aktivitas

Diagram aktivitas merupakan penggambaran aktivitas yang dilakukan oleh pengguna Sistem Informasi Surat Tugas, khususnya untuk seorang admin. Berikut adalah diagram aktivitas dari seorang admin yang ditunjukkan pada gambar 3.

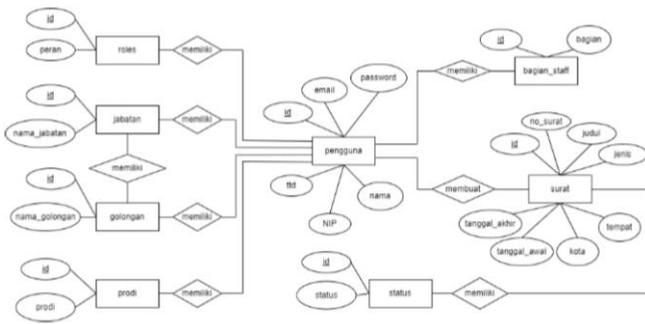


Gambar 3 Diagram Aktivitas Admin

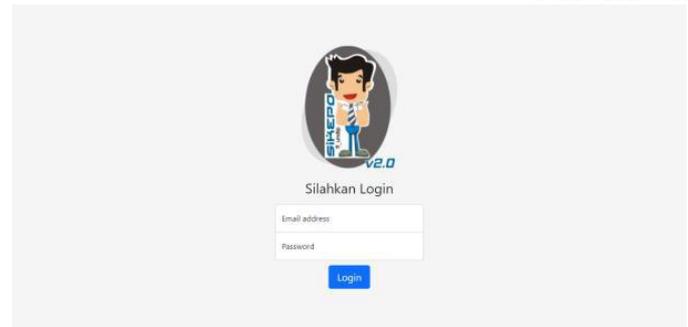
C. Perancangan Basis Data

1. ERD

ERD (*Entity Relationship Diagram*) merupakan salah satu metode pemodelan yang digunakan untuk menyusun suatu *database*. Gambar 4 merupakan gambaran ERD secara lengkap pada Sistem Informasi Surat Tugas meliputi entitas, atribut dan relasinya.



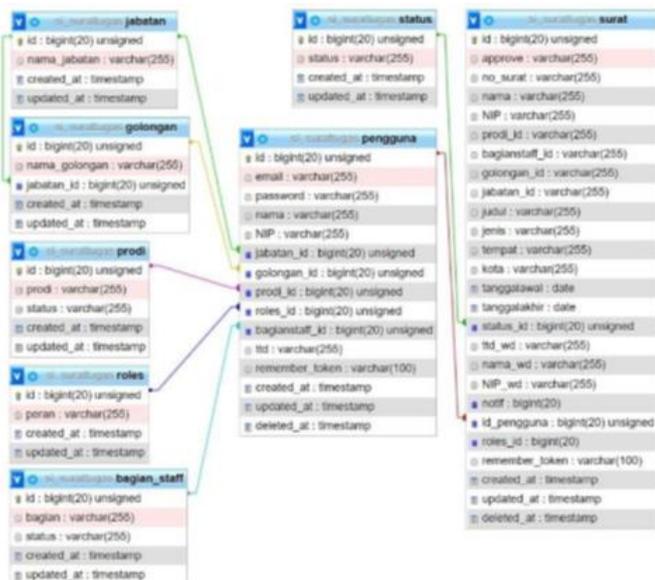
Gambar 4. ERD Sistem Informasi Surat Tugas



Gambar 12 Halaman Login

2. Class Diagram

Class diagram atau diagram kelas merupakan diagram menggambarkan struktur sistem dalam hal mendefinisikan kelas, kolom, atribut, tipe data, panjang data dan atribut kunci yang akan dihasilkan untuk membangun sistem. *Class Diagram* dari sistem ini dapat dilihat pada Gambar 5.



Gambar 5. Class Diagram

HASIL DAN PEMBAHASAN

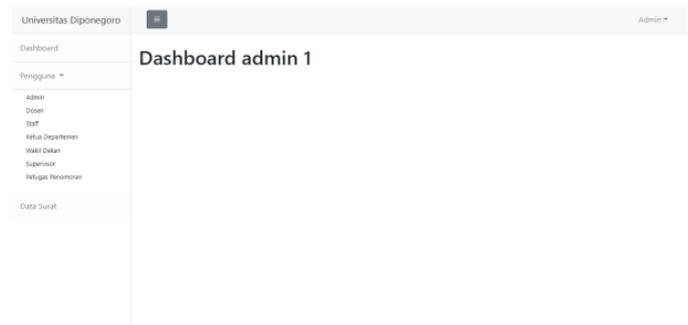
A. Implementasi Program

1. Tampilan Halaman Login

Halaman *login* merupakan halaman yang pertama kali ditampilkan kepada pengguna atau *user* ketika mengakses *website*. Pada halaman ini terdapat proses *input email* dan *password* untuk melakukan *login*. Pada halaman ini terdapat proses autentikasi untuk mengecek hak akses dari pengguna. Terdapat *guard* yang berfungsi sebagai *identifier* atau pengenal dari pengguna yang melakukan autentikasi. Tampilan dari halaman *login* dapat dilihat pada Gambar 6.

2. Tampilan Halaman Dashboard Admin

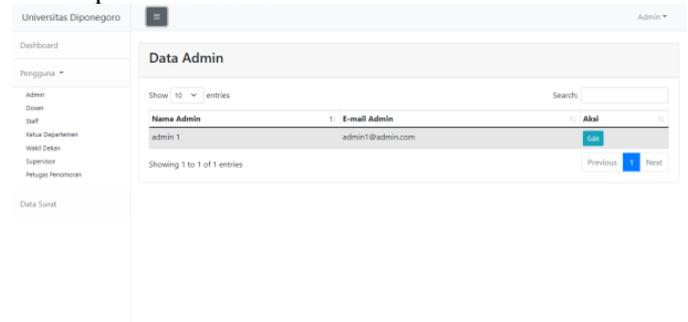
Halaman *Dashboard* merupakan halaman yang pertama kali ditampilkan kepada admin ketika berhasil melakukan proses *login* sebagai admin. Halaman *dashboard* admin ditunjukkan pada Gambar 7.



Gambar 7 Halaman Dashboard Admin

3. Tampilan Halaman Data Admin

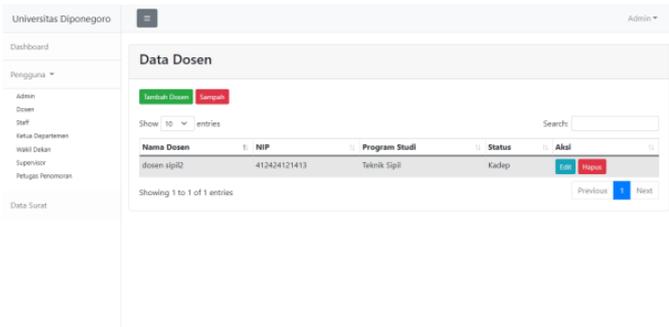
Halaman admin menampilkan data diri admin. Terdapat tombol "Edit" untuk mengubah data admin dan terdapat fitur *search* yang memudahkan admin untuk mencari data. Tampilan dari halaman data admin dapat dilihat pada Gambar 8.



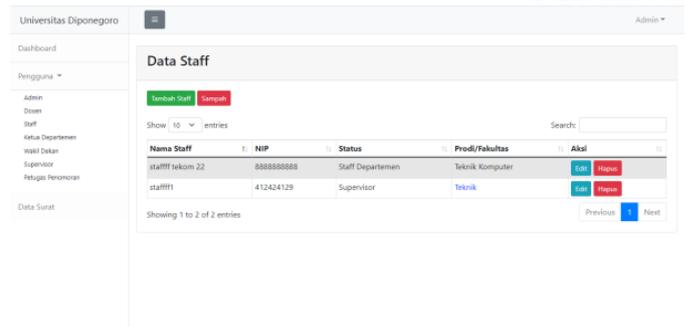
Gambar 8. Halaman Data Admin

4. Tampilan Halaman Data Dosen

Halaman dosen menampilkan seluruh data dosen yang telah ditambahkan oleh admin. Halaman data dosen ditunjukkan pada Gambar 9.



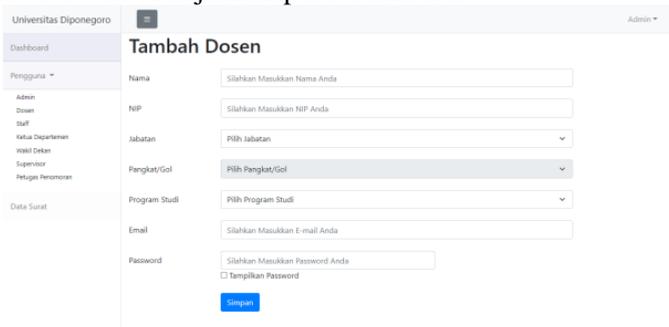
Gambar 9 Halaman Data Dosen



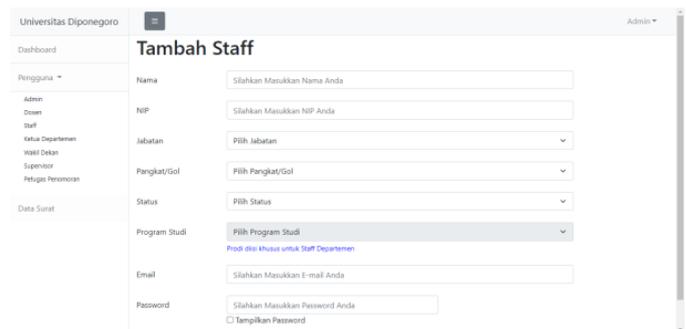
Gambar 12 Halaman Data Staf

Halaman tambah dosen berisi *form-form* untuk admin ketika ingin menambahkan data dosen. Halaman tambah data dosen ditunjukkan pada Gambar 10.

Halaman tambah staf berisi *form-form* untuk admin ketika ingin menambahkan data staf. Halaman tambah data staf ditunjukkan pada Gambar 13.



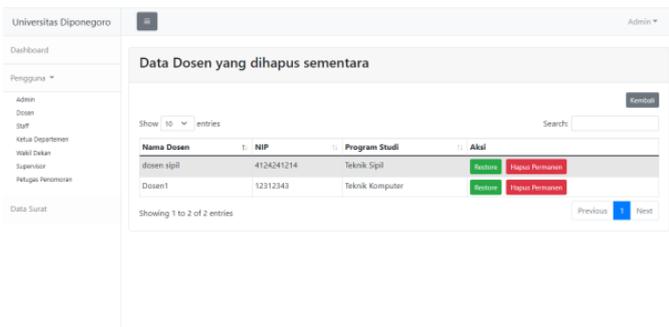
Gambar 10 Halaman Tambah Dosen



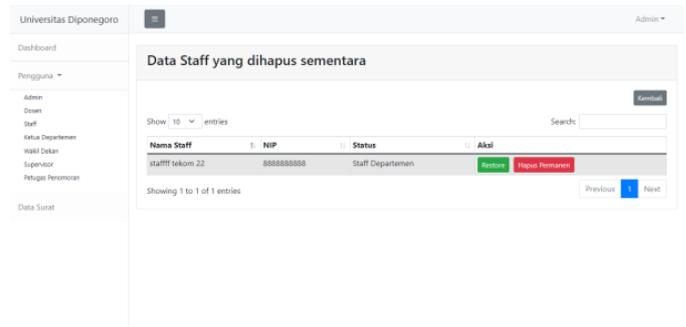
Gambar 13 Halaman Tambah Staf

Halaman data dosen sementara menampilkan data yang telah dihapus dari daftar data dosen. Halaman data dosen sementara ditunjukkan pada Gambar 11.

Halaman data staf sementara menampilkan data yang telah dihapus dari daftar data staf. Halaman data staf sementara ditunjukkan pada Gambar 14.



Gambar 11 Halaman Data Dosen Sementara



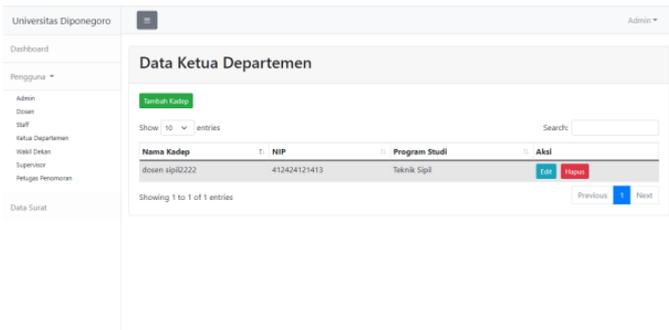
Gambar 14 Halaman Data Staf Sementara

5. Tampilan Halaman Data Staf

Halaman staf menampilkan seluruh data staf yang telah ditambahkan oleh admin. Halaman data staf ditunjukkan pada Gambar 12.

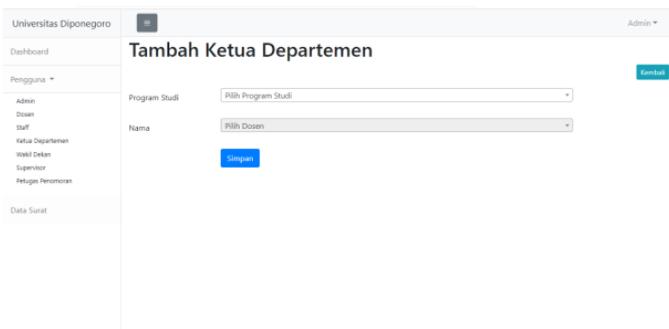
6. Tampilan Halaman Data Staf Sementara

Halaman data kadep menampilkan seluruh data Ketua Departemen yang telah ditambahkan oleh admin. Halaman data Ketua Departemen ditunjukkan pada Gambar 15.



Gambar 15 Halaman Data Ketua Departemen

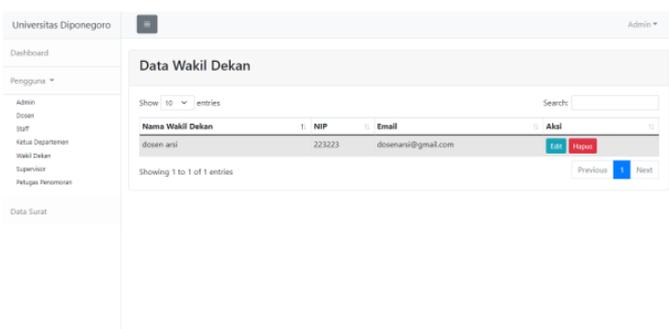
Halaman tambah ketua departemen berisi *form-form* untuk admin ketika ingin menambahkan data ketua departemen. Halaman tambah data ketua departemen ditunjukkan pada Gambar 16.



Gambar 16 Halaman Tambah Ketua Departemen

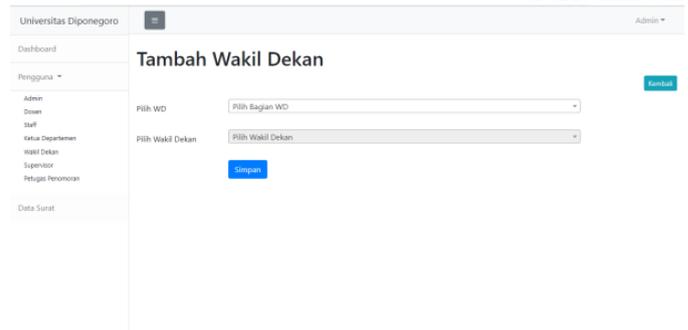
7. Tampilan Halaman Data Staf Sementara

Halaman wakil dekan menampilkan seluruh data wakil dekan yang telah ditambahkan oleh admin. Halaman data wakil dekan ditunjukkan pada Gambar 17.



Gambar 17 Halaman Data Wakil Dekan

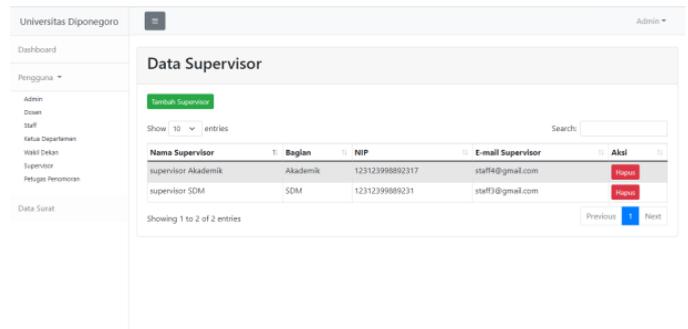
Halaman tambah wakil dekan berisi *form-form* untuk admin ketika ingin menambahkan data wakil dekan. Halaman tambah data wakil dekan ditunjukkan pada Gambar 18.



Gambar 18 Halaman Tambah Wakil Dekan

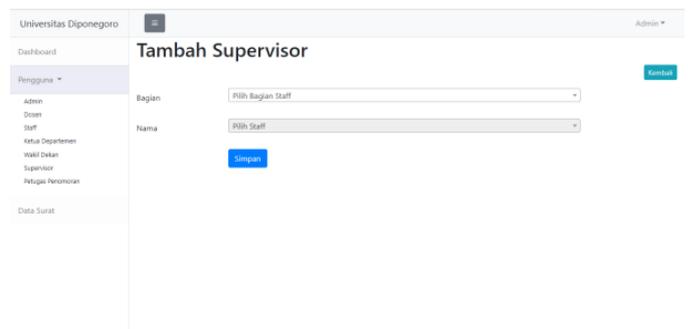
8. Tampilan Halaman Data Supervisor

Halaman *supervisor* menampilkan seluruh data *supervisor* yang telah ditambahkan oleh admin. Halaman data *supervisor* ditunjukkan pada Gambar 19.



Gambar 19 Halaman Data Supervisor

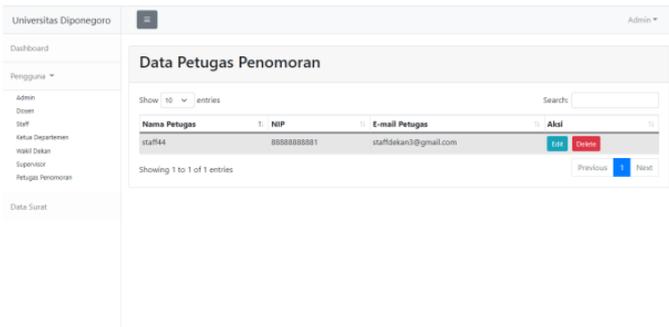
Halaman tambah *supervisor* berisi *form-form* untuk admin ketika ingin menambahkan data *supervisor*. Halaman tambah *supervisor* ditunjukkan pada Gambar 20.



Gambar 20 Halaman Tambah Supervisor

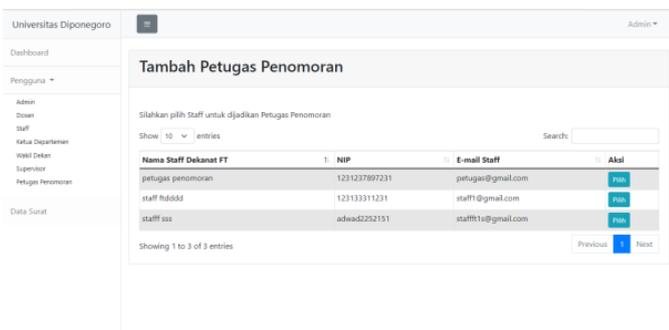
9. Tampilan Halaman Data Petugas Penomoran

Halaman petugas penomoran menampilkan seluruh data petugas penomoran yang telah ditambahkan oleh admin. Halaman data petugas penomoran ditunjukkan pada Gambar 21.



Gambar 21 Halaman Data Petugas Penomoran

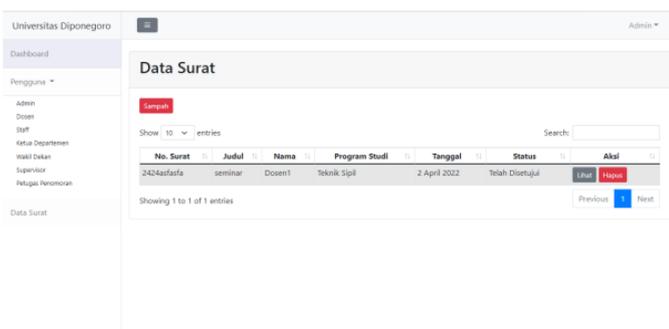
Halaman tambah petugas penomoran berisi *form-form* untuk admin ketika ingin menambahkan data *supervisor*. Halaman tambah petugas penomoran ditunjukkan pada Gambar 22.



Gambar 22 Halaman Tambah Petugas Penomoran

10. Tampilan Halaman Data Surat

Halaman data surat menampilkan data surat yang telah dibuat oleh dosen dan staf. Halaman data surat ditunjukkan pada Gambar 23.



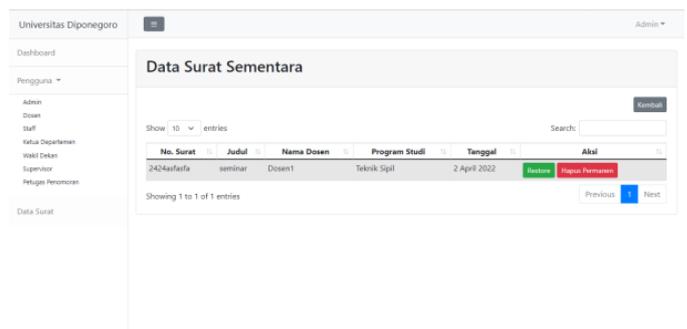
Gambar 23 Halaman Data Surat

Halaman lihat surat merupakan tampilan surat yang telah diubah menjadi pdf. Halaman lihat surat ditunjukkan pada Gambar 24.



Gambar 24 Halaman Lihat Surat

Halaman data surat sementara menampilkan data yang telah dihapus dari daftar data surat. Halaman data surat sementara ditunjukkan pada Gambar 25.



Gambar 25 Halaman Data Surat Sementara

B. Pengujian Sistem

Metode yang digunakan dalam menguji sistem aplikasi ini menggunakan metode *black box*. Tujuan dari metode *black box* ini adalah untuk menguji setiap fungsi dalam perangkat lunak agar dapat berjalan dengan baik.

1. Pengujian Halaman Masuk (*Login*)

Halaman *login* digunakan untuk autentikasi pengguna dan mengatur hak akses pengguna yang masuk ke sistem. Hasil pengujian pada halaman *login* ditunjukkan pada Tabel 1:

Pengujian pada halaman *login* dilakukan untuk menguji berjalannya fungsi autentikasi pengguna ketika masuk ke dalam sistem dan mengatur hak akses pengguna yang masuk. Hasil dari pengujian ini dapat dilihat pada Tabel 1.

Tabel 1 Pengujian Halaman Masuk (*Login*)

Pengujian	Bentuk Pengujian	Hasil yang diharapkan	Hasil
Pengujian buka sistem	Mengakses aplikasi atau sistem dengan menyetikkan url.	Menampilkan halaman <i>login</i>	Berhasil
Pengujian masuk ke sistem	Melakukan klik tombol " <i>login</i> "	Masuk ke halaman <i>dashboard</i>	Berhasil

		sesuai dengan hak akses	
--	--	-------------------------	--

2. Pengujian Halaman Data Admin

Pengujian pada halaman data admin dilakukan untuk menguji halaman dalam menampilkan informasi data admin. Hasil dari pengujian ini dapat dilihat pada Tabel 2.

Tabel 2 Pengujian Halaman Data Admin

Pengujian	Bentuk Pengujian	Hasil yang diharapkan	Hasil
pengujian halaman data admin	Mengakses menu <i>side bar</i> dengan submenu pengguna lalu mengklik menu "admin"	Menampilkan halaman data admin beserta isinya	Berhasil
masuk ke halaman edit profil	Mengakses melalui tombol "edit" yang ada di halaman data admin	Menampilkan <i>form-form</i> yang berisi data pengguna.	Berhasil
Pengujian <i>submit</i> perubahan data	Klik tombol "Simpan"	Menyimpan perubahan data pengguna ke <i>database</i>	Berhasil
Keluar dari sistem	Melakukan klik tombol "Logout"	Keluar dari sistem dan berhasil masuk ke halaman <i>login</i>	Berhasil

3. Pengujian Halaman Data Dosen

Pengujian halaman data dosen dilakukan untuk menguji fungsi-fungsi pada halaman, seperti menampilkan data dosen, menambah data dosen, mengubah data dosen dan menghapus data dosen. Hasil dari pengujian ini dapat dilihat pada Tabel 3.

Tabel 3 Pengujian Halaman Data Dosen

Pengujian	Bentuk Pengujian	Hasil yang diharapkan	Hasil
pengujian halaman data dosen	Mengakses menu <i>side bar</i> dengan submenu pengguna lalu mengklik menu "dosen"	Menampilkan halaman data dosen beserta isinya	Berhasil
Pengujian masuk ke halaman tambah dosen	Mengakses tombol "tambah" yang ada di halaman data dosen	Menampilkan halaman tambah dosen beserta <i>form-form</i> yang harus diisi	Berhasil

Pengujian <i>logout</i>	Klik tombol "Logout" di dalam <i>dropdown</i> pada <i>navbar</i>	Keluar dari sistem dan beralih ke halaman <i>login</i>	Berhasil
Pengujian <i>submit</i> pada halaman tambah dosen	Mengakses tombol "simpan"	Data dosen yang diisi berhasil di simpan pada <i>database</i> dan ditampilkan pada halaman data dosen	Berhasil
Pengujian masuk ke halaman edit dosen	Mengakses melalui tombol "edit" yang ada di halaman data dosen	Menampilkan halaman edit dosen beserta <i>form-form</i> yang sudah diisi.	Berhasil
Pengujian <i>submit</i> pada halaman edit dosen	Mengakses tombol "simpan" pada halaman edit dosen	Data dosen yang diubah berhasil diperbaharui pada <i>database</i> dan ditampilkan pada halaman data dosen	Berhasil
Pengujian aksi hapus	Mengakses tombol "hapus" pada halaman data dosen	Data dosen pada halaman data dosen terhapus, namun pada <i>database</i> data tetap ada. Pada bagian <i>delete_at</i> terisi karena menggunakan fitur <i>soft delete</i> .	Berhasil
pengujian masuk ke halaman data dosen sementara	Mengakses tombol "sampah" yang ada di halaman data dosen	Menampilkan halaman data dosen sementara akan terhapus dari <i>database</i>	Berhasil
Keluar dari sistem	Melakukan klik tombol "Logout"	Keluar dari sistem dan berhasil masuk ke halaman <i>login</i>	berhasil

4. Pengujian Halaman Data Staf

Pengujian pada halaman data staf dilakukan untuk menguji fungsi-fungsi pada halaman, seperti menampilkan data staf, menambah data staf, mengubah data staf, dan menghapus data staf. Hasil dari pengujian ini dapat dilihat pada Tabel 4.

Tabel 4 Pengujian Halaman Data Staf

Pengujian	Bentuk Pengujian	Hasil yang diharapkan	Hasil
pengujian halaman data staf	Mengakses menu <i>side bar</i> dengan submenu pengguna lalu mengklik menu “Staf”	Menampilkan halaman data staf beserta isinya	Berhasil
Pengujian masuk ke halaman tambah staf	Mengakses tombol “tambah” yang ada di halaman data staf	Menampilkan halaman tambah staf beserta <i>form-form</i> yang harus diisi	Berhasil
Pengujian submit pada halaman tambah staf	Mengakses tombol “simpan”	Data staf yang diisi berhasil di simpan pada <i>database</i> dan ditampilkan pada halaman data staf	Berhasil
Pengujian masuk ke halaman edit staf	Mengakses melalui tombol “edit” yang ada di halaman data staf	Menampilkan halaman edit staf beserta <i>form-form</i> yang sudah diisi.	Berhasil
Pengujian submit pada halaman edit staf	Mengakses tombol “simpan” pada halaman edit staf	Data staf yang diubah berhasil diperbaharui pada <i>database</i> dan ditampilkan pada halaman data staf	Berhasil
Pengujian aksi hapus	Mengakses tombol “hapus” pada halaman data staf	Data staf pada halaman data staf terhapus, namun pada <i>database</i> data tetap ada. Pada bagian <i>delete_at</i> terisi karena menggunakan fitur <i>soft delete</i> .	Berhasil
pengujian masuk ke halaman data staf sementara	Mengakses tombol “sampah” yang ada di halaman data dosen	Menampilkan halaman data staf sementara	Berhasil
Pengujian aksi <i>restore</i>	Mengakses tombol “ <i>restore</i> ” dan	Data dosen yang di <i>restore</i> berhasil	Berhasil

dan hapus permanen	tombol “Hapus Permanen” yang tersedia pada halaman data staf sementara	dikembalikan pada halaman data staf. Sedangkan Data staf yang di hapus secara permanen akan terhapus dari <i>database</i>	
Keluar dari sistem	Melakukan klik tombol “ <i>Logout</i> ”	Keluar dari sistem dan berhasil masuk ke halaman <i>login</i>	berhasil

5. Pengujian Halaman Data Ketua Departemen

Pengujian pada halaman ketua departemen dilakukan untuk menguji fungsi-fungsi pada halaman, seperti menampilkan data ketua departemen, menambah data, mengubah data, dan menghapus data ketua departemen. Hasil dari pengujian ini dapat dilihat pada Tabel 5.

Tabel 5 Pengujian Halaman Data Ketua Departemen

Pengujian	Bentuk Pengujian	Hasil yang diharapkan	Hasil
pengujian halaman data ketua departemen	Mengakses menu <i>side bar</i> dengan submenu pengguna lalu mengklik menu “Ketua Departemen”	Menampilkan halaman data ketua departemen beserta isinya	Berhasil
Pengujian masuk ke halaman tambah Ketua Departemen	Mengakses tombol “tambah” yang ada di halaman data Ketua Departemen	Menampilkan halaman tambah staf beserta <i>form-form</i> yang harus diisi	Berhasil
Pengujian submit pada halaman tambah Ketua Departemen	Mengakses tombol “simpan”	Data Ketua Departemen yang diisi berhasil disimpan, namun yang berubah hanya <i>roles_id</i> pada <i>database</i> menjadi angka 2(dua). Data yang telah ditambahkan ditampilkan pada halaman data Ketua Departemen	Berhasil
Pengujian aksi hapus	Mengakses tombol	Data Ketua Departemen	Berhasil

	“hapus” pada halaman data Ketua Departemen	pada halaman data Ketua Departemen terhapus, namun pada <i>database</i> data tetap ada. Pada bagian <i>roles_id</i> terisi dengan angka 1(satu).	
Keluar dari sistem	Melakukan klik tombol “Logout”	Keluar dari sistem dan berhasil masuk ke halaman login	berhasil

Pengujian aksi hapus	Mengakses tombol “hapus” pada halaman data Wakil Dekan	Data Wakil Dekan pada halaman data Wakil Dekan terhapus, namun pada <i>database</i> data tetap ada. Pada bagian <i>roles_id</i> terisi dengan angka 1(satu)	Berhasil
Keluar dari sistem	Melakukan klik tombol “Logout”	Keluar dari sistem dan berhasil masuk ke halaman login	berhasil

6. Pengujian Halaman Data Wakil Dekan

Pengujian pada halaman data wakil dekan dilakukan untuk menguji fungsi-fungsi pada halaman, seperti menampilkan data wakil dekan, menambah data, mengubah data, dan menghapus data. Hasil dari pengujian ini dapat dilihat pada Tabel 6.

Tabel 6 Pengujian Halaman Data Wakil Dekan

Pengujian	Bentuk Pengujian	Hasil yang diharapkan	Hasil
wakil dekan	Mengakses menu <i>side bar</i> dengan submenu pengguna lalu mengklik menu “wakil dekan”	Menampilkan halaman data wakil dekan beserta isinya	Berhasil
Pengujian masuk ke halaman tambah Wakil Dekan	Mengakses tombol “tambah” yang ada di halaman data Wakil Dekan	Menampilkan halaman tambah Wakil Dekan beserta isinya.	Berhasil
Pengujian submit pada halaman tambah Wakil Dekan	Mengakses tombol “pilih”	Data Wakil Dekan yang dipilih berhasil disimpan, namun yang berubah hanya <i>roles_id</i> pada <i>database</i> menjadi angka 3(tiga). Data yang telah ditambahkan ditampilkan pada halaman data Wakil Dekan	Berhasil

7. Pengujian Halaman Data Supervisor

Pengujian pada halaman data *supervisor* dilakukan untuk menguji fungsi-fungsi pada halaman, seperti menampilkan data *supervisor*, menambah data, mengubah data, dan menghapus data *supervisor*. Hasil dari pengujian ini dapat dilihat pada Tabel 7.

Tabel 7 Pengujian Halaman Data Supervisor

Pengujian	Bentuk Pengujian	Hasil yang diharapkan	Hasil
pengujian halaman data <i>supervisor</i>	Mengakses menu <i>side bar</i> dengan submenu pengguna lalu mengklik menu <i>supervisor</i>	Menampilkan halaman data <i>supervisor</i> beserta isinya	Berhasil
Pengujian masuk ke halaman tambah <i>supervisor</i>	Mengakses tombol “tambah” yang ada di halaman data <i>supervisor</i>	Menampilkan halaman tambah <i>supervisor</i> beserta <i>form-form</i> yang harus diisi	Berhasil
Pengujian submit pada halaman tambah <i>supervisor</i>	Mengakses tombol “pilih”	Data <i>supervisor</i> yang diisi berhasil disimpan, namun yang berubah hanya <i>roles_id</i> pada <i>database</i> menjadi angka 6(enam).	Berhasil
Pengujian aksi hapus	Mengakses tombol “hapus” pada halaman data <i>supervisor</i>	Data <i>supervisor</i> pada halaman data <i>supervisor</i> terhapus,	Berhasil

		namun pada database data tetap ada. Pada bagian <i>roles_id</i> terisi dengan angka 4(empat).	
Keluar dari sistem	Melakukan klik tombol "Logout"	Keluar dari sistem dan berhasil masuk ke halaman <i>login</i>	Berhasil

8. Pengujian Halaman Data Petugas Penomoran

Pengujian pada halaman data petugas penomoran dilakukan untuk menguji fungsi-fungsi pada halaman, seperti menampilkan, menambah, mengubah, dan menghapus data petugas penomoran. Hasil dari pengujian ini dapat dilihat pada Tabel 8.

Tabel 8 Pengujian Halaman Data Petugas Penomoran

Pengujian	Bentuk Pengujian	Hasil yang diharapkan	Hasil
halaman data Petugas Penomoran	Mengakses menu <i>side bar</i> dengan submenu pengguna lalu mengklik menu Petugas Penomoran	Menampilkan halaman data Petugas Penomoran beserta isinya	Berhasil
Pengujian masuk ke halaman tambah petugas penomoran	Mengakses tombol "tambah" yang ada di halaman data petugas penomoran	Menampilkan halaman tambah petugas penomoran beserta isinya.	Berhasil
Pengujian submit pada halaman tambah petugas penomoran	Mengakses tombol "pilih"	Data petugas penomoran yang dipilih berhasil disimpan, namun yang berubah hanya <i>roles_id</i> , pada database menjadi angka 7(tujuh).	Berhasil
Pengujian aksi hapus	Mengakses tombol "hapus" pada halaman data petugas penomoran	Data <i>supervisor</i> pada halaman data <i>supervisor</i> terhapus, namun pada database data tetap ada. Pada bagian <i>roles_id</i> terisi	Berhasil

		dengan angka 4(empat).	
Keluar dari sistem	Melakukan klik tombol "Logout"	Keluar dari sistem dan berhasil masuk ke halaman <i>login</i>	berhasil

9. Pengujian Halaman Data Surat

Pengujian pada halaman data surat dilakukan untuk menguji fungsi-fungsi pada halaman, seperti menampilkan data surat dan menghapus data surat. Hasil dari pengujian ini dapat dilihat pada Tabel 9.

Tabel 9. Pengujian Halaman Pesanan

Pengujian	Bentuk Pengujian	Hasil yang diharapkan	Hasil
pengujian halaman data surat	Mengakses menu <i>side bar</i> dengan mengklik menu data surat	Menampilkan halaman data surat beserta isinya	Berhasil
Pengujian aksi lihat	Mengakses tombol "lihat" pada halaman data surat	Melihat surat dalam bentuk pdf.	Berhasil
Pengujian aksi hapus	Mengakses tombol "hapus" pada halaman data surat	Data surat yang dipilih pada halaman data surat terhapus, namun pada database data tetap ada. Padabagian <i>delete_at</i> terisi karena menggunakan fitur soft delete.	Berh
Keluar dari sistem	Melakukan klik tombol "Logout"	Keluar dari sistem dan berhasil masuk ke halaman <i>login</i>	berhasil

10. Pengujian Halaman Ulasan

Pengujian pada halaman ini ditunjukkan dilakukan untuk menampilkan halaman ulasan dengan isi data yang sesuai. Hasil pengujian halaman ulasan ditunjukkan oleh Tabel 10.

Tabel 10. Pengujian Halaman Ulasan

Pengujian	Bentuk Pengujian	Hasil yang diharapkan	Hasil
Pengujian masuk ke	Mengakses melalui <i>button</i> "Tulis	Menampilkan halaman ulasan berupa	Berhasil

email pengguna yang berfungsi untuk membedakan data pengguna satu dengan pengguna lainnya.

DAFTAR PUSTAKA

- Damanik, B. (2021). Rancangan Sistem Informasi Smp Negeri 1 Tuhemberua Kabupaten Nias Utara Menggunakan Php Codeigniter. *JURNAL MAHAJANA INFORMASI*, 6(1), 6-15.
- Geniusa, A., & Samopa, F. (2013). Pembuatan Sistem Informasi Perjalanan Dinas Kantor Wilayah Direktorat Jenderal Perbendaharaan (SIPD-Kanwil DJPBN). *Jurnal Teknik ITS*, 2(2), A366-A370.
- Prabowo, I. H., & Kurniawan, A. (2020). Rancang Bangun Aplikasi Pengarsipan Surat Dan Disposisi (Apsd) Berbasis Website Menggunakan Framework Laravel (Studi Kasus: Kantor Dinas Pertanian Dan Ketahanan Pangan Kota Madiun). *Jurnal Manajemen Informatika*, 11(1).
- Widiyanto, W. W. (2018). Analisa Metodologi Pengembangan Sistem Dengan Perbandingan Model Perangkat Lunak Sistem Informasi Kepegawaian Menggunakan Waterfall Development Model, Model Prototype, Dan Model Rapid Application Development (Rad). *Jurnal Informa: Jurnal Penelitian Dan Pengabdian Masyarakat*, 4(1), 34-40.
- Wardana, S. H. (2010). *Menjadi Master PHP dengan Framework Codeigniter*. Elex Media Komputindo.
- Sari, D. P., Wijanarko, R., & Tengah, J. M. (2020). Implementasi Framework Laravel pada Sistem Informasi Penyewaan Kamera (Studi Kasus di Rumah Kamera Semarang). *J. Inform. dan Rekayasa Perangkat Lunak*, 2(1), 32.
- Banjarnahor, B. B., & Hartomo, K. D. (2016). Penerapan Laravel Framework Dalam Perancangan Sistem Informasi Promosi Produk Unggulan UKM Berbasis Web (Studi Kasus Dinas Perindustrian Perdagangan dan UMKM Kota Salatiga). *Universitas Kristen Satya Wacana Salatiga*, 1-27.
- Karnyoto, A. S. (2014). Analisis Perbandingan Optimasi Kecepatan Yii Framework dan Laravel. *Journal Dynamic Saint*, 1(2).
- Destiningrum, M., & Adrian, Q. J. (2017). Sistem Informasi Penjadwalan Dokter Berbassis Web Dengan Menggunakan Framework Codeigniter (Studi Kasus: Rumah Sakit Yukum Medical Centre). *Jurnal Teknoinfo*, 11(2), 30-37.

halaman ulasan	Ulasan” pada halaman pesanan	gambar produk, nama produk, bintang penilaian, komentar	
Pengujian <i>button</i> “Kirim”	Melakukan klik pada <i>button</i> “details”	Melakukan proses <i>input</i> pada <i>form</i> kemudian akan terjadi proses penampilan data	Berhasil

11. Pengujian Halaman Profil Pengguna

Pengujian pada halaman ini ditunjukkan dilakukan untuk menampilkan halaman profil dengan isi data yang sesuai. Hasil pengujian halaman profil pengguna ditunjukkan oleh Tabel 11.

Tabel 11. Pengujian Halaman Profil Pengguna

Pengujian	Bentuk Pengujian	Hasil yang diharapkan	Hasil
Pengujian masuk ke halaman profil pengguna	Mengakses melalui <i>button</i> “profil” di <i>dashboard</i> pengguna	Menampilkan halaman profil	Berhasil
Pengujian perubahan data pengguna	Melakukan klik <i>button</i> “Ubah Profil”	Melakukan proses <i>input</i> pada <i>form</i> kemudian akan terjadi proses perubahan data pengguna	Berhasil

KESIMPULAN

Berdasarkan pengujian dengan menggunakan metode *black-box*, dapat disimpulkan bahwa fungsi-fungsi setiap halaman yang ada pada sistem dapat berjalan dengan baik. Sistem Informasi Surat Tugas ini dapat membantu admin dalam membuat, mengubah dan menghapus data pengguna seperti data dosen dan data staf. Sistem Informasi Surat Tugas ini dapat membantu admin dalam mengelola data surat seperti menampilkan dan menghapus surat. Sistem Informasi Surat Tugas ini berhasil dikembangkan menggunakan struktur MVC (*Model, View, Controller*) yang terdapat pada kerangka kerja Laravel. Penggunaan *soft delete* pada sistem dapat meminimalisir kesalahan saat admin menghapus data pengguna. Terdapat fitur *validate unique* pada NIP dan