

Sistem Informasi Penjualan Toko Pupuk Surya Makmur

Dewantara Surya Jaya

¹Universitas Muhammadiyah Surakarta, Jl. A. Yani, Mendungan, Pabelan, Kec. Kartasura, Sukoharjo, Jawa Tengah 57162, Indonesia

✉ Email korespondensi: l200180013@student.ums.ac.id

Abstrak. Toko Pupuk khususnya Toko Pupuk Surya Makmur yang terletak di Jantir, Sindon, Ngemplak, Boyolali merupakan toko yang penting di pedesaan karena Petani adalah pekerjaan yang masih banyak dijumpai di sana. Toko Pupuk Surya Makmur terletak di daerah pedesaan yang penggunaan teknologi masih cukup minim. Aktivitas seperti mendata pembelian masih dilakukan secara manual menggunakan buku yang apabila terjadi kesalahan seperti salah tulis, salah hitung, dan sebagainya agak sulit untuk diperbaiki. Dengan pembuatan sistem informasi, diharapkan bisa mempermudah aktivitas-aktivitas seperti memasukkan data barang, memperbarui data barang, memasukkan data penjualan, dan lain-lain. Pengembangan sistem informasi ini menggunakan salah satu metode System Development Life Cycle (SDLC) yaitu Waterfall yang terdiri dari Analisis Kebutuhan, Desain, Pengembangan, Pengujian, dan Pemeliharaan. Sistem informasi ini berbasis localhost yang dibuat dengan text editor Visual Studio Code, Framework PHP yaitu Laravel, Framework CSS berupa Bootstrap, dan database MySQL. Untuk pengujiannya sendiri menggunakan System Usability Scale (SUS) dan Blackbox Testing untuk 10 responden calon Admin. Hasil pengujian mengungkap sistem mendapatkan skor rata-rata SUS sebesar 70 yang menandakan usability sistem ini tergolong Marginal dari segi kenyamanan, dan keefektifan dalam pemakaiannya.

Kata kunci: toko pupuk, sistem informasi, pengembangan.

PENDAHULUAN

Kemajuan teknologi yang sangat pesat dapat memberikan dampak positif dalam berbagai bidang seperti kesehatan, politik, ekonomi, pendidikan dan sebagainya. Untuk konteks bidang bisnis seperti usaha toko, adanya teknologi mempermudah aktivitas seperti pencatatan transaksi penjualan, pembuatan laporan, memperbarui stok barang, dan lain-lain. Teknologi yang saya maksud di sini adalah penggunaan sebuah sistem informasi.



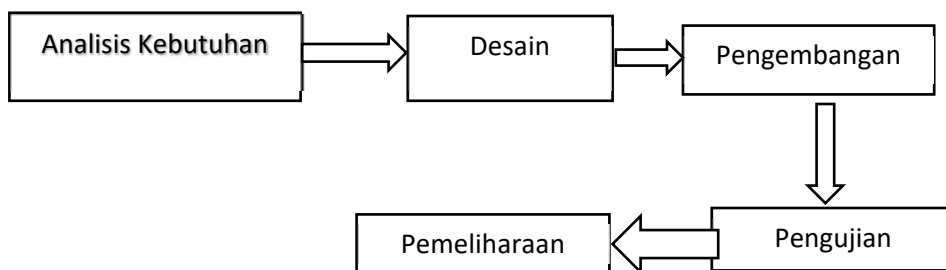
Tapi, sayangnya untuk toko-toko yang berada di pedesaan contohnya Toko Pupuk Surya Makmur masih belum memanfaatkan teknologi seperti itu. Toko Pupuk Surya Makmur masih menggunakan buku administrasi untuk mencatat secara manual. Hal ini kerap mengundang kesalahan dalam hal pencatatan seperti salah tulis, salah hitung, dan lain-lain. Dengan dibuatkan sistem informasi, diharapkan bisa meminimalisir kesalahan seperti itu.

Menurut [1] dalam penelitiannya membangun sistem informasi penjualan berbasis web yang bisa membantu proses pembelian di mana saja dengan proses yang lebih mudah dan cepat. Menurut [2] pembuatan sistem berbasis web dapat membantu pemilik untuk mengembangkan usahanya sehingga dapat diakses oleh siapapun, dimanapun, dan kapanpun. Menurut [3] sistem informasi dapat membantu kegiatan seperti pencatatan transaksi, pengelolaan stok menjadi lebih efisien. Menurut [4] web e-commerce dapat membantu pembeli dalam memesan produk secara online.

Kalau untuk sistem informasi yang akan saya kembangkan ini, akan memakai framework Laravel. Alasan saya memakai Laravel karena lebih mudah untuk memigrasi basis data dibandingkan pesaingnya, Codeigniter. Sistem ini bukan untuk jual beli pupuk dan alat pertanian, melainkan hanya untuk keperluan dokumentasi saja. Sistem ini hanya menggunakan tingkat pengguna Admin saja dan hanya dipakai di localhost komputer (tidak dijadikan web online).

METODE

Untuk membuat sistem informasi ini, menggunakan salah satu metode SDLC (*System Development Life Cycle*) yaitu *Waterfall*. Metode *waterfall* adalah siklus hidup tradisional mencakup metodologi pengembangan perangkat lunak yang menekankan proses yang logis dan sistematis pengujian sistem dilakukan untuk memastikan kinerja dan fungsionalitas yang baik [5] Metodenya sendiri dibagi menjadi beberapa tahapan berikut.



Gambar 1. Metode *Waterfall*



1. Analisis Kebutuhan Sistem

Analisis kebutuhan sistem merupakan tahapan yang paling krusial dalam metode *Waterfall* karena di tahap ini kita bisa mendeskripsikan apa saja yang dimiliki sistem dan bagaimana karakteristik sistem tersebut. Untuk analisis kebutuhan sistem ini, dibagi menjadi 2 yaitu Kebutuhan Fungsional dan non-Fungsional.

1.1 Kebutuhan Fungsional

Kebutuhan Fungsional sistem informasi toko pupuk ini meliputi :

Tabel 1. Kebutuhan Fungsional

No	Kebutuhan
1	Hanya Admin tingkat pengguna dari sistem informasi ini
2	Melihat tabel barang atau penjualan termasuk total pendapatan (untuk penjualan)
3	Menambahkan data (barang, penjualan, atau <i>invoice</i>)
4	Menampilkan detail data (barang, penjualan, atau <i>invoice</i>)
5	Memperbarui data (barang, penjualan)
6	Menghapus data suatu (barang, penjualan, atau <i>invoice</i>)
7	Menyortir data berdasarkan atribut (kecil ke besar, A ke Z, ataupun sebaliknya)
8	Mencari data suatu barang melalui kotak pencarian
9	Menyortir penjualan berdasarkan tanggal tertentu
10	Mencetak tabel ke Laporan PDF (barang, penjualan)

1.2 Kebutuhan Non-fungsional

No	Kebutuhan
1	Sistem berjalan dengan baik tanpa <i>bug</i> yang mengganggu
2	Dapat dibuka dengan <i>web browser</i> berupa Microsoft Edge, Google



	Chrome, Mozilla Firefox, dan lain-lain
--	--

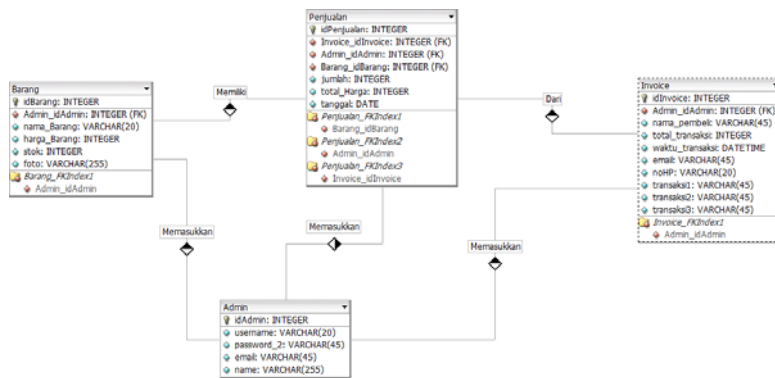
2 Desain

Desain atau istilah lainnya merancang merupakan lanjutan dari tahap Analisis Kebutuhan Sistem. Untuk sistem informasi, rancangan yang paling umum untuk dibuat adalah *Entity Relationship Diagram (ERD)*, *Use Case Diagram*, dan *Activity Diagram*.

2.1 Entity Relationship Diagram

Merupakan entitas interaksi antar data, diagram ini berdasarkan dengan kondisi yang sebenarnya pada basis data yang akan dibuat untuk sistem [6].

Ini adalah contoh ERD dari sistem informasi Toko Pupuk Surya Makmur yang bisa disimak.



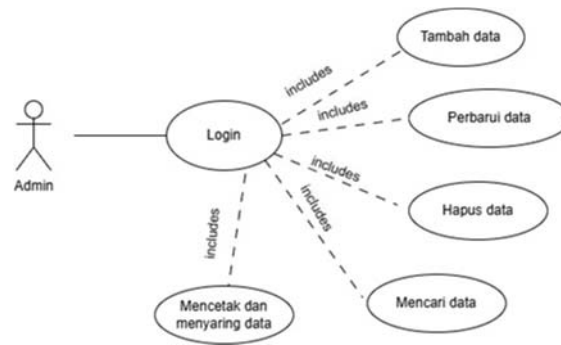
Gambar 2. Contoh ERD

Admin memasukkan data Barang, Penjualan, maupun Invoice. Barang memiliki *relationship one-to-many* dengan Penjualan. Begitu juga dengan Invoice memiliki hubungan *one-to-many* terhadap Penjualan.

2.2 Use Case Diagram

Use Case Diagram adalah merupakan pemodelan untuk perilaku sistem informasi yang akan dibuat [7]. *Use Case Diagram* dari sistem informasi ini dijelaskan pada gambar berikut. *Use Case Diagram* ini penting untuk menggambarkan apa saja yang dilakukan oleh berbagai tingkatan pengguna sistem informasi.





Gambar 3. Aktor dan Peranannya dalam *Use Case Diagram*

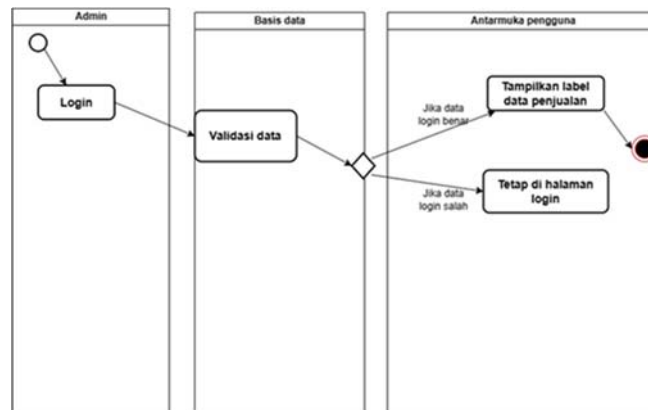
Keterangan:

- Tambah Data bisa dilakukan di entitas Barang, Penjualan, dan Invoice.
- Perbarui data dapat dilakukan di entitas Barang dan Penjualan. (Invoice tidak bisa)
- Hapus data (Barang, Penjualan, Invoice)
- Mencari data :
 - Barang (mencari berdasarkan semua atribut)
 - Penjualan (berdasarkan tanggal)
 - Invoice (berdasarkan nama Pembeli, No Hp, dan Email)
- Menyaring dan mencetak data (menyaring data kemudian di cetak ke bentuk pdf).

2.3 Activity Diagram

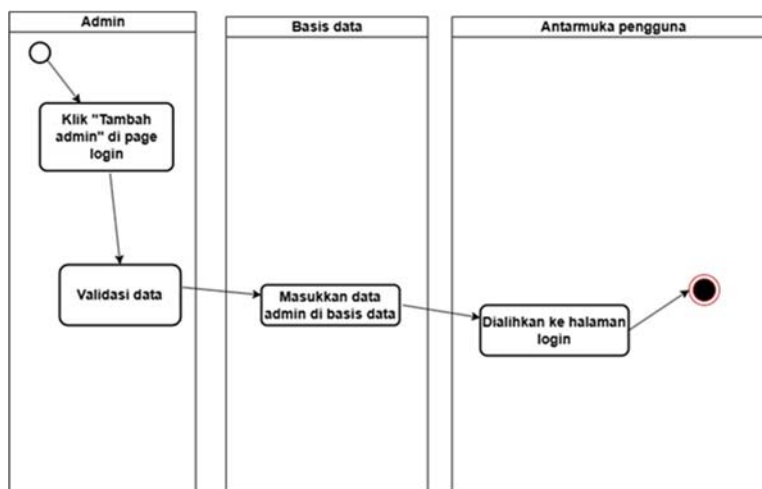
Activity diagram adalah diagram yang menggambarkan proses yang terjadi dalam suatu sistem [8]. Diagram ini penting sebagai gambaran detail dari suatu aktivitas dari permulaan sampai selesai. Contohnya dijelaskan pada gambar berikut.





Gambar 4. Aktivitas login admin

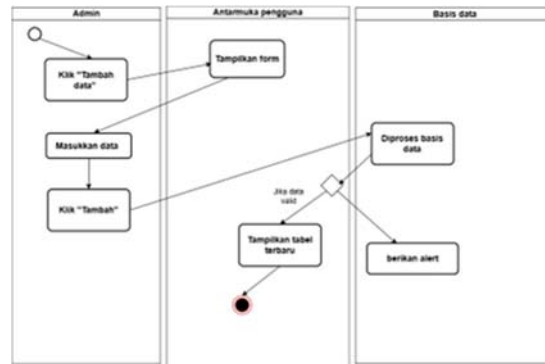
Admin bisa login dengan menggunakan data *e-mail* dan password. Bila datanya benar semua, langsung ditujukan pada halaman penjualan yang menampilkan tabel penjualan, Bila datanya ada yang salah, maka tetap di halaman login sambil menampilkan pesan *error*.



Gambar 5. Menambahkan admin baru

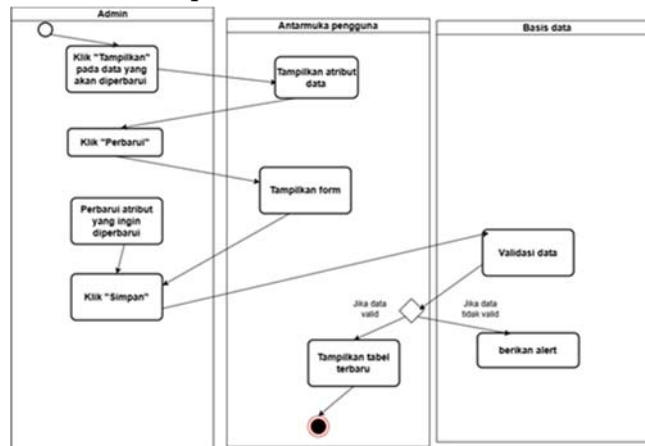
Pertama, klik tombol "Tambah Admin" di halaman login. Untuk menambahkan admin baru, diperlukan untuk mengisi data berupa nama, email, username, dan password ke form register. Setelah memasukkan semua itu, anda bisa login ke halaman utama.





Gambar 6. Aktivitas menambahkan data

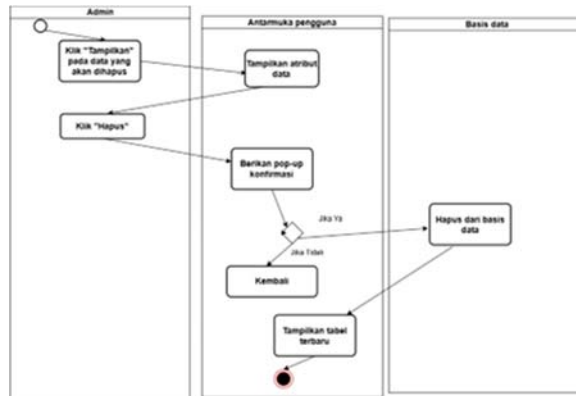
Admin bisa mengklik tombol “Tambah data” maka akan ditampilkan laman form yang berisi data-data yang harus diisi. Bila sudah selesai memasukkan datanya, klik tombol “Tambah” untuk menambahkan data baru dalam tabel. Bila semua masukannya *valid*, maka di *redirect* ke tabel terbaru. Bila ada kolom yang masukkan datanya tidak *valid*, berikan alert pada kolom tersebut.



Gambar 7. Alur memperbaiki data

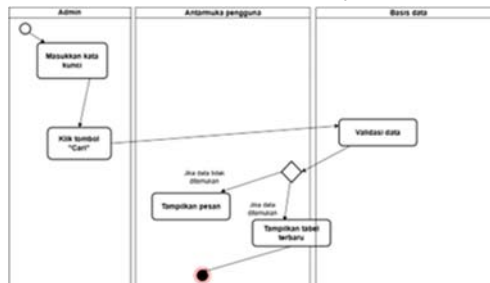
Pada baris data yang ingin diperbarui, klik “Tampilkan” maka akan *direct* ke halaman detail atribut baris tersebut. Untuk memperbaiki, kita bisa klik “Perbarui”. Selanjutnya prosesnya seperti kita memasukkan baris data baru perbedaannya hanya mengganti masukkan atribut yang ingin kita ganti.





Gambar 8. Rincian alur menghapus data

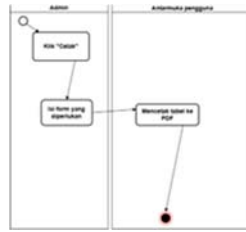
Pada baris data yang ingin dihapus, klik "Tampilkan" maka akan dibawa ke halaman detail atribut baris tersebut. Untuk menghapus baris data tersebut, bisa klik "Hapus" maka akan ada *pop-up* konfirmasi apakah anda benar-benar ingin menghapus baris data tersebut. Jika anda klik "Ya" pada konfirmasi maka data tersebut akan terhapus dari *database* dan anda dibawa ke tabel terbaru setelahnya.



Gambar 9. Proses pencarian data

Perhatikan dulu *placeholder* pada kolom pencarian supaya tidak salah memasukkan kata kunci misal kolom pencarian memiliki *placeholder* "Masukkan tanggal" maka anda harus memasukkan bagian dari tanggal misal angka tahun, bulan, maupun hari. Masukkan kata kunci pada kolom pencarian lalu tekan tombol *enter* pada *keyboard* perangkat. Jika tabel dengan kata kunci tersebut ditemukan, maka tampilkan tabel tersebut. Jika tidak, berikan pesan "Tidak ada data.."





Gambar 10. Cetak laporan atau tabel ke PDF

Klik tombol “Cetak” pada tabel yang ingin dicetak. Lalu isi berdasarkan apa tabel tersebut ingin dicetak ke pdf. Misalkan, mencari data barang yang harganya 50.000 hingga 70.000. Kalau ingin mencetak keseluruhan data tabel, tinggalkan kosongkan saja *form*-nya. Langkah terakhir, klik “Cetak”.

3. Pengembangan

Rancangan tersebut kemudian dibuatkan kode program yang akan menjadi program komputer. Untuk pengkodean *front end* menggunakan *framework* CSS yaitu Bootstrap + Javascript + Font awesome (untuk *icon*) sedangkan *back end* menggunakan *framework* PHP yaitu Laravel, *virtual server* lokal menggunakan Xampp dan terakhir untuk pengkodean memakai Visual Studio Code.

4. Pengujian

Setelah melalui tahap Pengembangan, tentunya harus dilakukan Pengujian untuk dapat memastikan sistem bekerja dengan baik. Jenis pengujian ada 2 yaitu pengujian sistem dan pengujian pengguna. Untuk pengujian sistem, menggunakan metode *Blackbox Testing*, adalah tahap pengujian untuk menilai apakah sistem berfungsi semestinya atau tidak [9] . Dalam pengujian *Blackbox* di sistem ini, penulis akan menggunakan teknik *Equivalence Partitioning*. Dalam teknik tersebut, pengembang harus membagi pengujian masukkan form menjadi dua kondisi, yaitu kondisi masukkan *valid* dan kondisi masukkan tidak *valid*. Jadi, untuk kedua kondisi tersebut, harus ada *Test Case* masing-masing.

Sementara untuk pengujian pengguna digunakan metode *System Usability Scale* (SUS), untuk mengetahui tingkat *usability* dari suatu sistem.

Langkah-langkah yang perlu dilakukan :

4.1 *Blackbox Testing* dengan teknik *Equivalence Partitioning*

1. Menyiapkan beberapa *Test Case* yang mana harus ada *Test Case* yang



menghasilkan masukkan *valid* dan *Test Case* yang menghasilkan masukkan tidak *valid*.

2. Menguji *Test Case* tersebut
3. Jika ada hasil *Test Case* yang tidak diharapkan, pengembang bisa memperbaiki struktur kode programnya (*coding*).

4.2. System Usability Scale

1. Memberikan beberapa pertanyaan kuesioner yang biasa digunakan dalam SUS

Tabel 2. Contoh pernyataan atau kuesioner dalam SUS

No	Pernyataan
1	Saya akan sering menggunakan sistem ini.
2	Saya tahu kalau sistemnya rumit.
3	Sistem ini mudah untuk digunakan.
4	Saya membutuhkan bantuan orang lain/ahli untuk bisa menggunakan sistem ini.
5	Berbagai fungsi di sistem ini terhubung dengan baik.
6	Terlalu banyak inkonsistensi dalam sistem ini.
7	Menurut saya, banyak orang bisa belajar menggunakan sistem ini dengan sangat cepat.
8	Saya pikir sistemnya terlalu sulit digunakan.
9	Saya nyaman dalam menggunakan sistem ini.
10	Saya harus belajar banyak hal sebelum menggunakan sistem ini

2. Siapkan bobot nilai misalkan Sangat Setuju itu 5, Setuju itu 4, Ragu-ragu 3, Tidak setuju 2 dan Sangat Tidak Setuju 1
3. Mencari suara dari 10 calon pengguna atau Admin sebagai responden
4. Hitung nilai SUS caranya pada pernyataan ganjil nilainya dikurangi 1, pernyataan genap kurangi nilainya dari lima, lalu tambahkan nilai pertanyaan genap dan ganjil. Langkah terakhir, tinggal kali dengan 2.5 untuk mendapatkan nilai SUS.
5. Hitung rata-rata nilai SUS yang didapat kelima responden lalu nilai rata-rata SUS bisa diklasifikasikan melalui tabel berikut



Tabel 3. Klasifikasi Nilai SUS

Skor SUS	Arti Skor
0 – 50.9	<i>Not Acceptable</i>
51 – 70.9	<i>Marginal</i>
71 - 100	<i>Acceptable</i>

Hasil kedua tes tersebut akan menentukan seberapa sesuai sistem tersebut dengan calon pengguna dan seberapa bagus fungsionalitasnya.

5. Pemeliharaan

Tahap terakhir adalah pemeliharaan. Pemeliharaan adalah usaha dari pengembang untuk menjaga dan meningkatkan kualitas dari sistem yang mereka kembangkan. Pemeliharaan bisa dilakukan dengan cara memperbaiki *bug* yang ada, membuat sistem lebih efisien, menambahkan fitur dan lain-lain. Tujuan pemeliharaan ini adalah untuk meningkatkan kepuasan pengguna seiring perbaikan dan perawatan yang dilakukan.

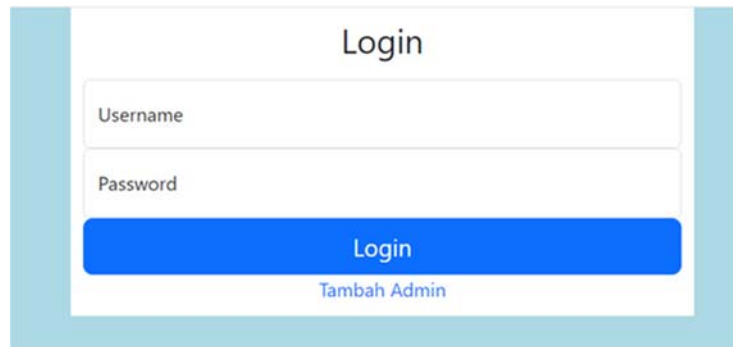
HASIL

1. Antarmuka pengguna dan *Blackbox Testing*

Berikut tampilan antarmuka pengguna dari sistem informasi penjualan toko pupuk ini.

1.1 Pengujian Form Login





Gambar 11. Login

Seperti form *login* pada umumnya, ada 2 kolom yaitu Username dan Password lalu satu tombol untuk *Login*. Jika salah satu Username atau Password salah saat *Login*, akan ditampilkan peringatan “Login gagal”. Jika kedua Username dan Password benar saat *Login*, maka akan *redirect* ke Tab atau *dashbord* Penjualan.

Tabel 4. Pengujian Form Login

No	Test Case	Hasil yang diharapkan	Hasil pengujian
1	Mengisi username dan password dengan data yang benar lalu klik “Login”	<i>Login</i> berhasil dan ditampilkan <i>dashboard</i> penjualan	<i>Login</i> berhasil dan ditampilkan <i>dashboard</i> penjualan
2	Mengisi username benar dan password salah lalu klik “Login”	Peringatan “Login gagal”	Peringatan “Login gagal”
3	Mengisi username salah dan password benar lalu klik “Login”	Peringatan “Login gagal”	Peringatan “Login gagal”



1.2 Pengujian Form Tambah Admin



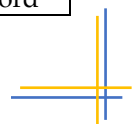
Gambar 12. Menambah admin

Validasi masukkan untuk form “Tambah Admin” sebagai berikut.

1. Nama, minimal 3 karakter (wajib diisi).
2. Email, format dns, tidak boleh duplikat (wajib diisi).
3. Username, Minimal 3 karakter tidak boleh duplikat (wajib diisi).
4. Password, Minimal 5 karakter (wajib diisi).

Tabel 5. Pengujian Form Tambah Admin

No	Test Case	Hasil yang diharapkan	Hasil pengujian
1	Memasukkan semua input dengan format yang benar	Diarahkan menuju form <i>login</i>	Diarahkan menuju form <i>login</i>
2	Memasukkan nama dengan format yang salah (kurang dari 3 karakter)	Peringatan “Username minimal 3 karakter”	Peringatan “Username minimal 3 karakter”
3	Memasukkan email dengan format yang salah	Peringatan “Format email salah”	Peringatan “Format email salah”
4	Memasukkan email yang sudah dipakai	Peringatan “Email sudah dipakai”	Peringatan “Email sudah dipakai”
5	Memasukkan username dengan format yang salah username	Peringatan “username minimal 3 karakter”	Peringatan “username minimal 3 karakter”
6	Memasukkan username yang sudah dipakai	Peringatan bahwa username sudah dipakai	Peringatan “username sudah dipakai”
7	Memasukkan password	Peringatan bahwa password	Peringatan “Password



	dengan format yang salah	harus minimal 5 karakter	minimal 5 karakter"
--	--------------------------	--------------------------	---------------------



1.3 Pengujian Tab Penjualan (Kolom Pencarian)



Gambar 13. Tab Penjualan

Di Halaman Penjualan, Admin dapat melihat data penjualan yang baru dimasukkan, melihat *detail* penjualan, total pendapatan per halaman dan keseluruhan.

Tombol “Tambah Data” untuk menuju halaman form tambah data penjualan dan Tombol “Filter” untuk menuju halaman form filter dan cetak data penjualan

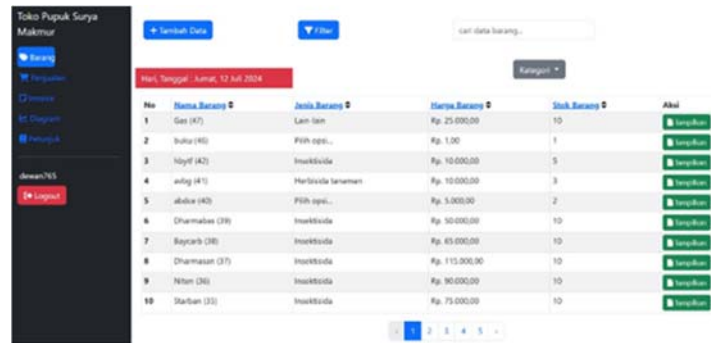
Kolom Pencarian di Tab Penjualan dapat digunakan untuk mencari data penjualan berdasarkan nama barang atau tanggal penjualan. Untuk mencari berdasarkan tanggal penjualan bisa memakai format yyyy-mm-dd (tahun 4 digit-bulan 2 digit-hari).

Tabel 6. Pengujian Kolom Pencarian di Halaman Penjualan

No	Test Case	Hasil yang diharapkan	Hasil pengujian
1	Mencari data penjualan berdasarkan nama barang yang ada di <i>database</i>	Di tampilkan hasil pencarian berdasarkan kata kunci	Sistem merespon dengan menampilkan hasil tabel pencarian berdasarkan kata kunci
2	Mencari data penjualan berdasarkan tanggal yang ada di <i>database</i>	Di tampilkan hasil pencarian berdasarkan kata kunci (data tanggal penjualan)	Sistem merespon dengan menampilkan hasil tabel pencarian berdasarkan kata kunci
3	Mencari data penjualan yang tidak ada di <i>database</i>	Sistem menampilkan semacam peringatan bahwa data tidak ditemukan	<i>Body</i> tabel diganti dengan peringatan “Tidak ada data...”



1.4 Pengujian Tab Barang (Kolom Pencarian)



Gambar 14. Tab Barang

Kolom Pencarian di Tab Barang dapat digunakan untuk mencari data barang berdasarkan nama barang, jenis barang, dan, harga barang.

Tabel 7. Pengujian Kolom Pencarian di Tab Barang

No	Test Case	Hasil yang diharapkan	Hasil pengujian
1	Mencari data barang berdasarkan nama barang yang ada di <i>database</i>	Di tampilkan tabel hasil pencarian berdasarkan kata kunci	Sistem merespon dengan menampilkan hasil tabel pencarian berdasarkan kata kunci
2	Mencari data barang berdasarkan jenis barang yang ada di <i>database</i>	Di tampilkan tabel hasil pencarian berdasarkan kata kunci	Sistem merespon dengan menampilkan hasil tabel pencarian berdasarkan kata kunci
3	Mencari data barang berdasarkan harga barang yang ada di <i>database</i>	Di tampilkan tabel hasil pencarian berdasarkan kata kunci	Sistem merespon dengan menampilkan hasil tabel pencarian berdasarkan kata kunci
4	Mencari data barang dengan kata kunci yang tidak ada di <i>database</i>	Sistem menampilkan semacam peringatan data tidak ditemukan	<i>Body</i> tabel diganti dengan peringatan "Tidak ada data..."



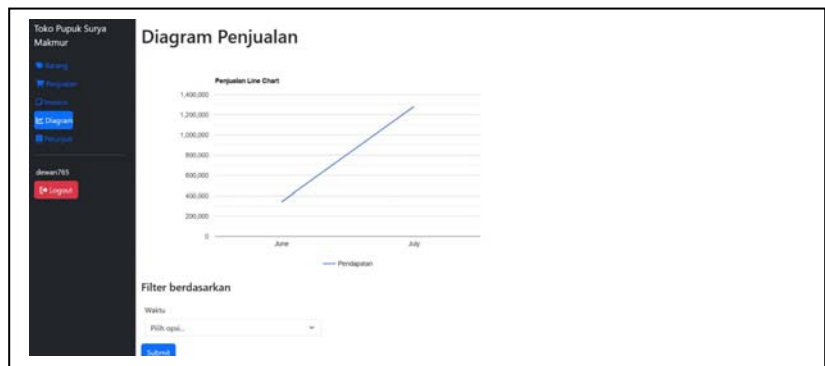
1.5 Pengujian Tab Invoice (Kolom Pencarian)



Gambar 15. Tab Invoice

Kolom Pencarian di Tab Invoice dapat dipakai untuk mencari data berdasarkan nama pembeli, email, dan no HP.

Tabel 8. Pengujian Kolom Pencarian di Tab Invoice



Gambar 16. Tab Diagram

Di Tab Diagram, Admin bisa melihat *line chart* penjualan yang bisa disaring berdasarkan hari, bulan, maupun tahun.



Gambar 17. Tab Petunjuk




Tab Petunjuk lebih ke cara penggunaan sistem ini.

No	Test Case	Hasil yang diharapkan	Hasil pengujian
1	Mencari data invoice berdasarkan nama pembeli yang ada di <i>database</i>	Di tampilkan tabel hasil pencarian berdasarkan kata kunci	Sistem merespon dengan menampilkan hasil tabel pencarian berdasarkan kata kunci karena invoice dengan nama pembeli menyerupai kata kunci ada di <i>database</i>
2	Mencari data invoice berdasarkan email yang ada di <i>database</i>	Di tampilkan tabel hasil pencarian berdasarkan kata kunci	Sistem merespon dengan menampilkan hasil tabel pencarian berdasarkan kata kunci karena invoice dengan email menyerupai kata kunci ada di <i>database</i>
3	Mencari data invoice berdasarkan no HP yang ada di <i>database</i>	Di tampilkan tabel hasil pencarian berdasarkan kata kunci	Sistem merespon dengan menampilkan hasil tabel pencarian berdasarkan kata kunci karena invoice dengan no HP menyerupai kata kunci ada di <i>database</i>
4	Mencari data invoice dengan kata kunci yang tidak ada di <i>database</i> kata kunci : 586	Sistem menampilkan semacam peringatan data tidak ditemukan	<i>Body</i> tabel diganti dengan peringatan "Tidak ada data..."



1.6 Pengujian Tab Barang (Form Tambah Data Barang)



Gambar 18. Tambah data barang

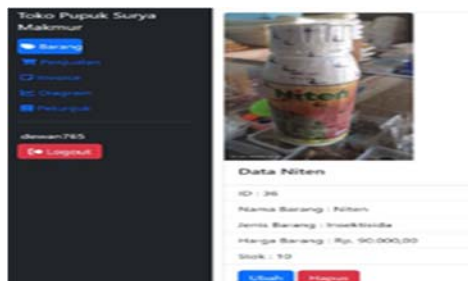
Validasi masukkan untuk form tambah data barang sebagai berikut.

1. Nama Barang, minimal 2 karakter (wajib diisi)
2. Jenis Barang, wajib pilih salah satu
3. Harga, tidak boleh 0 (wajib diisi)
4. Stok, wajib diisi
5. Foto, opsional dengan maksimal ukuran file 4 MB



Tabel 9. Pengujian Form Tambah Data Barang

No	Test Case	Hasil yang diharapkan	Hasil pengujian
1	Memasukkan inputan dengan format yang benar	Notifikasi Data barang berhasil diinputkan ke database dan ditampilkan tabel barang terbaru setelah masukkan	Notifikasi data barang berhasil ditambahkan dan tabel barang terbaru ditampilkan dengan baris terbaru berada di paling awal
2	Memasukkan input nama barang dengan format yang salah	Sistem memberikan peringatan nama barang minimal 2 karakter	Sistem menolak input dan menampilkan peringatan "Nama Barang minimal 2 karakter"
3	Tidak memilih jenis barang	Sistem memberikan peringatan untuk memilih jenis barang	Sistem menolak input dan menampilkan peringatan "Mohon pilih jenis barang"
4	Memasukkan input harga dengan format yang salah	Sistem memberikan peringatan harga barang tidak boleh 0	Sistem menolak input dan menampilkan peringatan "Harga tidak boleh 0"
5	Memasukkan foto dengan ukuran yang lebih besar dari 4 MB	Sistem memberikan peringatan bahwa ukuran foto tidak boleh lebih dari 4 MB	Sistem menolak input dan memberikan peringatan "Ukuran file maksimal 4 MB"

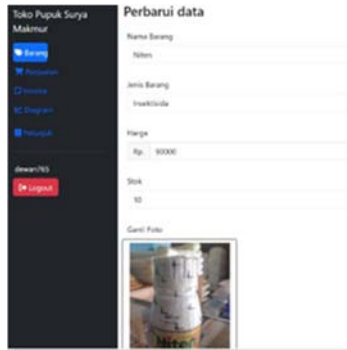


Gambar 19. Tampilkan data barang



Halaman ini menampilkan data barang berupa ID, Nama Barang, Jenis Barang, Harga, dan Stok dalam sebuah *card*. Tombol Ubah dan Hapus juga ada dalam *card* tersebut. Untuk menampilkannya, anda bisa klik “Tampilkan” pada baris data yang ingin ditampilkan.

1.7 Pengujian Tab Barang (Form Perbarui Data Barang)



Gambar 20. Perbarui data barang


Untuk daftar validasi input untuk memperbarui data barang, sama dengan validasi untuk form tambah data barang.

Tabel 10. Pengujian Form Perbarui Data Barang

No	Test Case	Hasil yang diharapkan	Hasil pengujian
1	Memasukkan inputan dengan format yang benar	Notifikasi Data barang berhasil diperbarui di database dan ditampilkan tabel barang terbaru setelah masukkan	Notifikasi data barang berhasil diperbarui dan ditampilkan tabel barang terbaru dengan baris terbaru berada di paling awal



1.8. Pengujian Tab Penjualan (Form Tambah Data Penjualan)



Gambar 21. Tambah data penjualan

Validasi untuk form tambah data penjualan sebagai berikut.

1. Jumlah barang, tidak boleh 0 (wajib diisi)
2. Total harga, didapat dari harga * jumlah barang yang dipilih
3. Tanggal, dengan format yyyy-mm-dd (di *database*).

Untuk data penjualan ini penulis menghubungkannya dengan data Barang dengan *one-to-many relationship* dengan *foreign key* barang_id.

Tabel 11. Pengujian Form Tambah Data Penjualan

No	Test Case	Hasil yang diharapkan	Hasil pengujian
1	Memasukkan inputan dengan format yang benar	Notifikasi Data penjualan berhasil diinputkan ke database dan ditampilkan tabel penjualan terbaru setelah masukkan	Notifikasi data penjualan berhasil ditambahkan dan tabel penjualan terbaru ditampilkan dengan baris terbaru berada di paling awal
2	Memasukkan input jumlah dengan format yang salah	Sistem memberikan peringatan bahwa jumlah barang tidak boleh 0	Sistem menolak input dan memberikan peringatan "jumlah barang tidak boleh 0 atau barang habis"

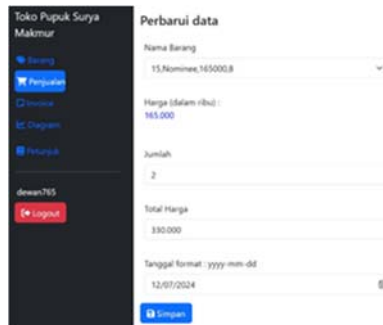




Gambar 22. Tampilkan data penjualan

Untuk menampilkan data penjualan satu barang seperti pada gambar 22, anda bisa klik “Tampilkan” pada baris data penjualan yang dikehendaki untuk ditampilkan. Ada tombol “Ubah” untuk mengubah nilai data penjualan dan “Hapus” untuk menghapus data penjualan tersebut.

1.9 Pengujian Tab Penjualan (Form Perbarui Data Penjualan)



Gambar 23. Ubah data penjualan

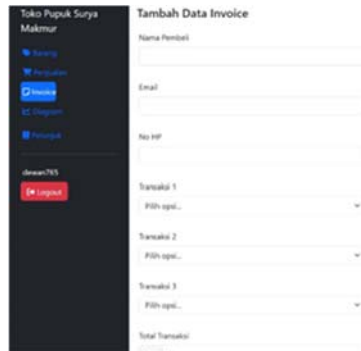
Validasi untuk form perbarui data penjualan sama dengan validasi form tambah data penjualan.

Tabel 12. Pengujian Form Perbarui Data Penjualan

No	Test Case	Hasil yang diharapkan	Hasil pengujian
1	Memasukkan inputan dengan format yang benar	Notifikasi Data penjualan berhasil diperbarui di database dan ditampilkan tabel barang terbaru setelah masukkan	Notifikasi data penjualan berhasil diperbarui dan tabel penjualan terbaru ditampilkan dengan baris terbaru berada di paling awal



1.10 Pengujian Tab Invoice (Form Tambah Data Invoice)



Gambar 24. Tambah data Invoice

Validasi untuk form tambah data Invoice sebagai berikut.

1. Nama Pembeli, minimal 3 karakter (wajib diisi)
2. Email, harus memiliki @ dan domain (.com, .id, dsb) (opsional)
3. No HP, minimal 9 karakter angka (opsional)
4. Transaksi 1, 2, dan 3 tidak boleh ada yang sama, kecuali ada yang kosong (wajib mengisi salah satu)
5. Total transaksi didapat dari total harga ketiga transaksi



Tabel 13. Pengujian Form Tambah Data Invoice

No	Test Case	Hasil yang diharapkan	Hasil pengujian
1	Memasukkan inputan dengan format yang benar.	Notifikasi Data invoice berhasil diinputkan ke database dan ditampilkan tabel invoice terbaru setelah masukkan	Notifikasi "Data invoice james berhasil ditambahkan" dan tabel invoice terbaru ditampilkan dengan baris terbaru berada di paling awal
2	Memasukkan input nama pembeli dengan format yang salah nama	Sistem memberikan peringatan nama pembeli minimal 3 karakter	Sistem menolak input dan menampilkan peringatan "Nama pembeli minimal 3 karakter"
3	Memasukkan email dengan format yang salah	Sistem memberikan peringatan untuk memasukkan email dengan format yang benar	Sistem menolak input dan menampilkan peringatan "format email tidak valid"
4	Memasukkan No HP dengan format yang salah	Sistem memberikan peringatan no HP minimal 9 karakter	Sistem menolak input dan menampilkan peringatan "No hp minmal 9 karakter"
5	Transaksi 1,2,3 ada yang sama	Sistem memberikan peringatan bahwa transaksi tidak boleh ada yang sama	Sistem menolak input dan memberikan peringatan bahwa transaksi ada yang sama





No	Nama Barang	Jumlah Barang	Total Harga
1	Nominee	1	Rp. 165.000,00
2	Nugrass	1	Rp. 100.000,00
3			Rp. 0,00
Total Transaksi			Rp. 265.000,00

Gambar 25. Tampilkan data Invoice

Halaman ini menampilkan data Invoice berupa Data Pembeli (Id *Invoice*, Nama Pembeli, Email, No HP, dan waktu transaksi) dan Data Transaksi (Barang apa saja yang dibeli dan Total Transaksi). Tombol Hapus berada di bawah Tabel data transaksi.

1.11 Pengujian Tab Barang (Form Filter dan Cetak Data Barang)



Gambar 26. Filter Data barang

Validasi untuk form filter dan cetak data barang sebagai berikut.

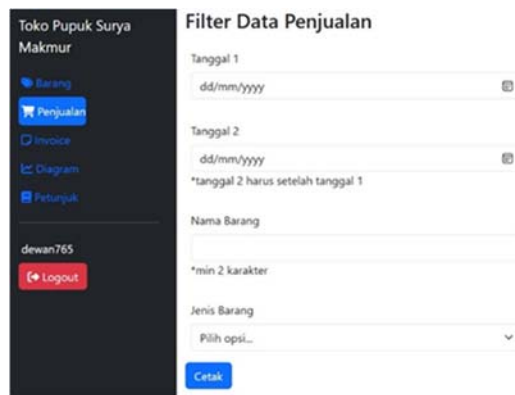
1. Harga 1, harus diisi angka
2. Harga 2, harus diisi angka (harus lebih besar atau sama dengan harga 1)
3. Stok barang 1, harus diisi angka
4. Stok barang 2, harus diisi angka (harus lebih besar atau sama dengan stok barang 1)



Tabel 14. Pengujian Form Filter dan Cetak Data Barang

No	Test Case	Hasil yang diharapkan	Hasil pengujian
1	Memasukkan inputan dengan format yang benar dan data hasil filternya tersedia	Sistem menerima input dan mencetak file pdf berisi tabel hasil filter data barang	File pdf berisi laporan tabel hasil filter data berhasil dicetak atau diunduh
2	Memasukkan inputan dengan format yang benar dan data hasil filternya kosong.	Sistem menerima input dan mencetak file pdf berisi tabel yang kosong	File pdf berisi laporan tabel yang kosong dengan peringatan "Tidak ada data..." berhasil dicetak atau diunduh

1.12 Pengujian Tab Penjualan (Form Filter dan Cetak Data Penjualan)



Toko Pupuk Surya Makmur

- Barang
- Penjualan**
- Invoice
- Diagram
- Petunjuk

dewan765

Logout

Filter Data Penjualan

Tanggal 1
dd/mm/yyyy

Tanggal 2
dd/mm/yyyy
*tanggal 2 harus setelah tanggal 1

Nama Barang
*min 2 karakter

Jenis Barang
Pilih opsi...

Cetak

Gambar 27. Filter Data penjualan



Tabel 15. Pengujian Form Filter dan Cetak Data Penjualan

No	Test Case	Hasil yang diharapkan	Hasil pengujian
1	Memasukkan inputan dengan format yang benar dan data hasil filturnya tersedia. [1]	Sistem menerima input dan mencetak file pdf berisi tabel hasil filter data penjualan	File pdf berisi laporan tabel hasil filter data berhasil dicetak atau diunduh
2	Memasukkan inputan dengan format yang benar dan data hasil filturnya kosong.	Sistem menerima input dan mencetak file pdf berisi tabel yang kosong	File pdf berisi laporan tabel yang kosong dengan peringatan "Tidak ada data..." berhasil dicetak atau diunduh

2. Pengujian Pengguna Menggunakan SUS (*System Usability Scale*)

Karena tingkat atau *role* pengguna di sistem ini hanya seorang admin, maka responden yang penulis cari sebanyak 10 orang. Seluruh responden tersebut mencoba menggunakan sistem dan mengisi kuesioner yang penulis sebar. Hasil kuesioner tersebut akan direkapitulasi dan dihitung nilai rata-ratanya dan berikutnya diukur karakter dari masing-masing variabel *system usability scale*.



Tabel 16. Hasil Pengujian SUS

Responden	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	Jumlah	Skor
R1	3	1	4	3	3	3	4	3	3	2	29	72.5
R2	1	3	4	3	3	1	4	3	3	3	28	70
R3	3	3	3	3	3	3	3	3	3	3	30	75
R4	3	3	3	3	3	3	3	3	3	1	28	70
R5	3	3	3	3	3	2	3	3	3	1	27	67.5
R6	3	3	3	1	3	1	3	3	3	1	24	60
R7	3	3	3	1	3	3	3	3	3	1	26	65
R8	3	3	4	3	3	3	3	3	3	1	29	72.5
R9	3	3	3	3	4	3	3	3	3	2	30	75
R10	3	3	3	3	3	3	3	3	3	2	29	72.5
Rata-rata skor SUS											700/10 = 70	

Rata-rata skor SUS adalah 70 yang menandakan sistem tersebut masuk dalam kategori *Marginal* menurut Tabel 3. Artinya sistem ini sudah termasuk rata-rata dalam segi kepuasan dan keefektifan pemakaiannya.

KESIMPULAN

Pembuatan Sistem Informasi Penjualan Toko Pupuk Surya Makmur ini dimaksudkan untuk menggantikan pengelolaan administrasi penjualan di Toko Pupuk Surya Makmur yang sebelumnya masih menggunakan buku. Sistem informasi adalah website berbasis framework Laravel. Fitur-fitur didalamnya berisi sistem CRUD data barang atau penjualan, mencari data barang atau penjualan, grafik penjualan dan invoice pembelian. Sistem ini dikembangkan berdasarkan model *Waterfall* yang meliputi analisis kebutuhan sistem, desain, pengembangan, pengujian, dan pemeliharaan. Saat pengujian Blackbox, sistem ini telah terancang dan berfungsi dengan baik menu-menu maupun tombol-



tombolnya. Dalam pengujian SUS pun mendapatkan skor rata-rata 70 yang sudah termasuk kategori Marginal untuk kenyamanan penggunaannya.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada dosen pembimbing Bapak Nurgiyatna dan para responden yang bersedia meluangkan waktunya untuk mengisi kuesioner.

DAFTAR PUSTAKA

- [1] L. Permatasari and Y. Suhari, "Sistem Informasi Penjualan Pupuk Bersubsidi Dan Nonsubsidi Berbasis Web," *J. Sains Komput. Inform.*, vol. 14, no. 1, pp. 1–6, 2023.
- [2] S. Suharnawi, R. R. Sani, and W. P. Loka, "Sistem Informasi Penjualan Online Berbasis Web pada Toko Sari Tani Tegal," *JOINS (Journal Inf. Syst.*, vol. 5, no. 2, pp. 256–264, Nov. 2020, doi: 10.33633/joins.v5i2.3853.
- [3] A. Rafidah, P. S. Informatika, F. Komunikasi, D. A. N. Informatika, and U. M. Surakarta, "Sistem informasi manajemen penjualan toko pertanian sahabat tani," 2020.
- [4] B. A. Metalya and Alhani, "Sistem Informasi Penjualan Tanaman dengan Framework Codeigniter, Studi Kasus: Toko Tani Makmur," *Pros. Semin. Ilm. Sist. Inf. Dan Teknol. Inf.*, vol. Vol.XII, no. No.1, pp. 314–326, 2023.
- [5] Y. Anggraini, R. Fadillah, and N. T. Suban, "Perancangan Sistem Informasi Persediaan Obat Pada Klinik Medika Prima Berbasis Web Menggunakan Metode Waterfall," *BINER J. Ilmu Komputer, Tek. dan Multimed.*, vol. 1, no. 2, pp. 87–98, 2023, [Online]. Available: <https://www.journal.mediapublikasi.id/index.php/Biner/article/view/2855/1354>
- [6] A. Puspaningrum and E. Sudarmilah, "Sistem Informasi Manajemen Peminjaman (Studi Kasus: Pengelolaan Aset Dan Tata Ruang Taman Budaya Jawa Tengah)," *Technol. J. Ilm.*, vol. 11, no. 1, p. 37, 2020, doi: 10.31602/tji.v11i1.2699.
- [7] D. Hardiyanti, "Sistem Informasi Sekolah Berbasis Web pada Sekolah Dasar Negeri (SDN) Seriti," *Indones. J. Educ. Humanit.*, vol. 1, no. 3, pp. 156–168, 2021, [Online]. Available: <http://ijoehm.rcipublisher.org/index.php/ijoehm/article/view/28>
- [8] S. Rokhmah and I. C. Utomo, "Rancang Bangun Aplikasi Penjualan Pada Toko Tepat Sukoharjo," pp. 486–494, 2021.
- [9] S. Shofia and D. A. Anggoro, "Sistem Informasi Manajemen Administrasi Dan Keuangan Pada Tk-It Permata Hati Sumberrejo-Bojonegoro," *JITK (Jurnal Ilmu Pengetah. dan Teknol. Komputer)*, vol. 5, no. 2, pp. 221–230, 2020, doi: 10.33480/jitk.v5i2.1192.

