

Pembuatan Sistem Informasi Admin Berbasis Web pada Aplikasi Pemesanan Buket Menggunakan Pustaka ReactJS

Web-Based Admin Information System Development for Bouquet Ordering Application Using ReactJS Library

Muhammad Ridwan^{*,1)}, Rizal Isnanto²⁾, Erwin Adriono³⁾

^{1,2,3)} Program Studi Teknik Komputer, Fakultas Teknik, Universitas Diponegoro
Jl. Prof. Soedarto, SH, Kampus Undip Tembalang, Semarang, Indonesia 50275

How to cite: M. Ridwan, R. Isnanto, and E. Adriono, "Pembuatan Sistem Informasi Admin Berbasis Web pada Aplikasi Pemesanan Buket Menggunakan Pustaka ReactJS", *Jurnal Teknik Komputer*, Vol. 2, No. 3, pp. 190-196, Oct 2023, doi: 10.14710/jtk.v2i3.38881 [Online].

Abstract – *Bucket SOC is a business that provides bouquet making services located in Surakarta. This business creates a web-based admin information system to optimize the bouquet ordering service. The system is used by the admin to manage products, banners, categories, and orders. The admin information system created for the Bucket SOC business uses ReactJS library and the Waterfall method. The website uses the Javascript programming language. The research steps include literature study, consumer needs analysis, website design, black box testing, and final project report writing. The website provides a list of bouquet products for sale and other information needed by users or potential buyers. The results of the study show that the Bucket SOC admin information system was successfully designed using ReactJS and the Waterfall method. The system can run well according to black box testing.*

Keywords – *Website, Admin, JavaScript, ReactJS library, Waterfall method*

Abstrak – *Bucket SOC adalah usaha jasa buket bunga di Surakarta yang membuat sistem informasi admin web untuk mengoptimalkan layanan pemesanan. Sistem ini, menggunakan pustaka ReactJS dan metode Waterfall, membantu admin mengelola produk buket, banner, kategori, dan transaksi. Situs web yang menggunakan Javascript menyediakan daftar produk buket dan informasi lain untuk pengguna atau pembeli. Langkah penelitian adalah studi literatur, analisis kebutuhan konsumen, perancangan situs web, pengujian kotak-hitam, dan penulisan laporan akhir. Hasil penelitian menunjukkan bahwa sistem informasi admin Bucket SOC berhasil dirancang dan diuji. Sistem ini berfungsi dengan baik sebagai media pengelolaan usaha.*

Kata kunci – *Situs Web, Admin, Javascript, pustaka ReactJS, metode Waterfall*

^{*}) Penulis Korespondensi (Muhammad Ridwan)
Email: muhammadridwan@students.undip.ac.id

I. PENDAHULUAN

Penggunaan teknologi Internet dan aplikasi perangkat lunak telah memberikan kemajuan dalam aspek kehidupan, termasuk bisnis. Sistem jual beli secara daring menjadi salah satu hasil dari kemajuan teknologi tersebut. Bagi usaha "Bucket SOC" yang bergerak dibidang jasa pembuatan buket. Proses penjualan yang digunakan sebelumnya yang memanfaatkan media sosial Instagram dan Whatsapp dinilai masih kurang efektif karena proses transaksi masih dilakukan secara manual.

Pembuatan sistem informasi admin berbasis web pada aplikasi pemesanan buket diharapkan dapat mengoptimalkan layanan pemesanan buket. Sistem informasi tersebut akan digunakan oleh admin untuk mengelola produk, banner, kategori pada produk yang dijual, dan transaksi yang masuk. Selain itu, sistem informasi admin berbasis web ini juga akan memudahkan admin dalam mengakses data pelanggan dan melakukan pencatatan transaksi secara otomatis

Dengan menggunakan teknologi informasi yang tepat, bisnis dapat menghemat biaya dan waktu, serta meningkatkan kualitas layanan yang diberikan kepada pelanggan. Pengembangan sistem informasi admin berbasis web pada usaha "Bucket SOC" merupakan langkah yang tepat dalam memanfaatkan teknologi informasi untuk meningkatkan efektivitas dan efisiensi bisnis. Sistem informasi admin tersebut akan membantu mengelola bisnis dengan lebih baik, mempercepat proses transaksi, dan memberikan pengalaman berbelanja yang lebih baik bagi pelanggan.

II. TINJAUAN LITERATUR

Penelitian terdahulu adalah proses mencari dan mengumpulkan penelitian sebelumnya yang terkait dengan masalah yang sedang dibahas, kemudian digunakan sebagai referensi dan dasar untuk penelitian saat ini. Dalam penelitian ini, penelitian terdahulu digunakan sebagai bahan studi pustaka dalam penelitian yang sedang dilakukan.

Menurut Huda dkk [1] dalam penelitiannya *e-ticketing* dianggap sebagai solusi praktis dan efisien untuk mengatasi masalah dalam penjualan tiket acara musik di wilayah Lampung. Hal ini memiliki kesamaan dalam pemanfaatan aplikasi yang dikembangkan. Dengan adanya sistem informasi admin diharapkan penjual dapat melakukan pengelolaan produk berbasis Internet, dimana data yang dikelola disimpan pada sistem penyimpanan *cloud* sehingga dapat diakses dimana saja. Pada aplikasi ini diharapkan pelayanan terhadap pembeli meningkat dan fasilitas yang diberikan untuk kemudahan penjual dalam mengelola produk dan pesanan menjadi lebih efektif dan efisien.

Menurut Panjaitan dkk [2] dalam penelitiannya tentang sistem *e-reporting* yang dirancang membantu perusahaan dalam melakukan pelaporan keuangan dan manajemen menggunakan ReactJS dan Firebase. Penggunaan teknologi ReactJS dalam pengembangan sistem diharapkan dapat mempercepat pembuatan aplikasi karena efisiensi dan kecepatan yang disediakan. Pada ReactJS menyediakan sistem bersifat modular dan skalabilitas dimana komponen yang dibuat dapat digunakan kembali sehingga memudahkan dalam pengembangan dan perawatan. Komunitas yang kuat dan sumber atau yang disediakan cukup lengkap sehingga proses penyelesaian cukup menghemat waktu. Kemudian Firebase sendiri menyediakan pengelolaan yang terintegrasi dimana terdapat layanan *hosting* dan basisdata, serta didukung dengan kemudahan penggunaan produk google lain. Firebase juga memiliki autentikasi dan otorisasi sehingga aplikasi web dijamin keamanannya..

III. METODE PENELITIAN

Metode *Waterfall* adalah model pengembangan perangkat lunak linier dan terstruktur dengan tahapan yang jelas dan urutan yang tidak dapat diubah. Setiap fase dimulai setelah fase sebelumnya selesai, dan meliputi analisis kebutuhan, perancangan, implementasi, pengujian, dan pemeliharaan. Metode ini cocok bagi tim pengembang yang membutuhkan struktur dan pengendalian ketat [3].

Tahapan dalam melakukan metode *Waterfall*

1. Analisis Kebutuhan (*Requirement Analysis*)

Pada metode *Waterfall*, kebutuhan pengguna merupakan hal yang dilakukan sebelum melakukan pengembangan perangkat lunak. Dengan mengetahui informasi dengan beberapa cara seperti observasi, data yang diperoleh nantinya akan digunakan sesuai kebutuhan pengguna akan aplikasi yang dikembangkan.

2. Perancangan (*Design*)

Data yang diperoleh pada tahap analisis kebutuhan akan diolah untuk menentukan desain pengembangan. Desain ini nantinya akan digunakan untuk menentukan gambaran lengkap apa yang harus dikerjakan serta membantu menyiapkan kebutuhan perangkat keras atau perangkat lunak dalam pembuatan arsitektur aplikasi yang akan dibuat.

3. Implementasi (*Implementation*)

Implementasi merupakan tahapan dimana rancangan yang sudah dibuat di implementasikan dalam kode sumber dan dites secara berkala.

4. Pengujian (*Testing*)

Pengujian dilakukan untuk memastikan aplikasi berfungsi sesuai dengan spesifikasi.

5. Pemeliharaan (*Maintenance*)

Setelah perangkat lunak selesai dikembangkan dan dioperasikan oleh pengguna, tahap pemeliharaan dilakukan oleh pengembang untuk memperbaiki kesalahan yang mungkin tidak terdeteksi pada tahap sebelumnya. Tahap ini memungkinkan perangkat lunak untuk terus berfungsi secara optimal seiring waktu dan memastikan kepuasan pengguna yang tinggi.

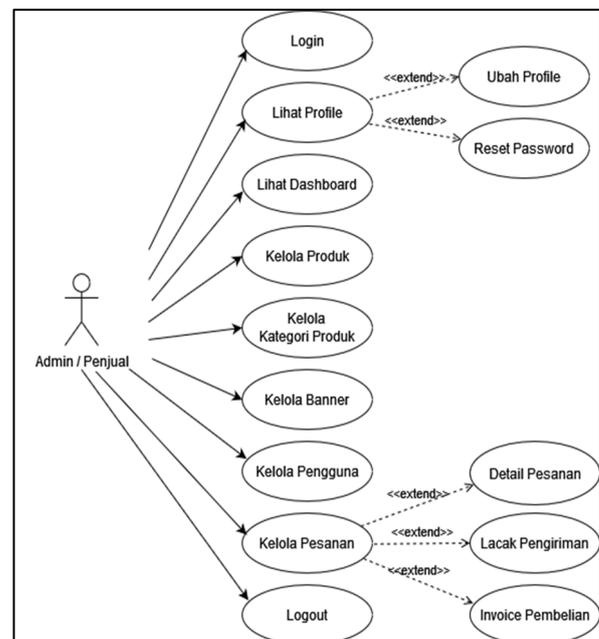
IV. HASIL DAN PEMBAHASAN

Bab ini menjabarkan tentang tahap perancangan, implementasi dan pengujian sistem.

A. Perancangan sistem

1. Diagram *Use Case*

Diagram *use case* diagram visual yang menggambarkan interaksi antara aktor dan fungsi-fungsi sistem dalam suatu sistem. Diagram ini memberikan pemahaman tentang skenario-skenario dan tindakan-tindakan yang dilakukan oleh aktor serta respons yang diberikan oleh sistem [4]. Hubungan antara pengguna yaitu sebagai admin terhadap sistem yang akan dikembangkan dapat digambarkan melalui Diagram *use case* yang dapat ditunjukkan pada Gambar 1



Gambar 1. Diagram *Use Case* admin atau penjual

Skenario dalam pemesanan produk yang dilakukan oleh pengguna dapat ditunjukkan pada Tabel 1.



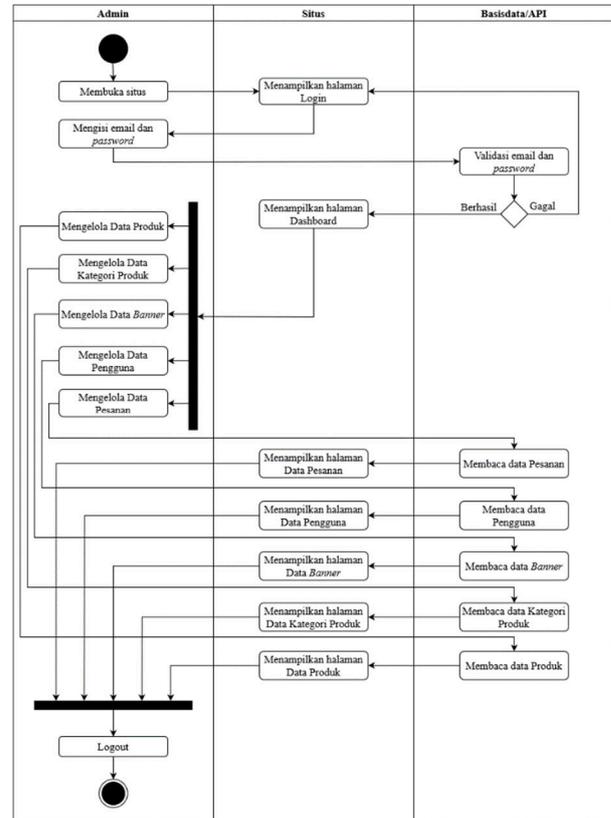
Tabel 1. Deskripsi Diagram *Use Case*

No	Use Case	Deskripsi
1.	Login	Merupakan proses memasukkan identitas pengguna agar dapat masuk ke sistem admin
2.	Lihat Profil	Merupakan proses melihat detail informasi dari pengguna admin yang login.
3.	Ubah Profil	Merupakan proses mengubah detail informasi dari pengguna admin yang login
4.	Reset Password	Merupakan proses <i>me-reset password</i> pengguna admin yang digunakan untuk login.
5.	Lihat Dashboard	Merupakan proses melihat informasi penjualan dan data pada menu sistem informasi admin
6.	Kelola Produk	Merupakan proses mengelola produk untuk menambahkan produk baru, merubah informasi produk yang sudah ada, dan menghapus produk yang tidak digunakan.
7.	Kelola Kategori Produk	Merupakan proses mengelola kategori untuk menambahkan kategori baru, merubah informasi kategori yang sudah ada, dan menghapus kategori yang tidak digunakan.
8.	Kelola Banner	Merupakan proses mengelola <i>banner</i> untuk menambahkan <i>banner</i> baru, merubah informasi <i>banner</i> yang sudah ada, dan menghapus <i>banner</i> yang tidak digunakan.
9.	Kelola Pengguna	Merupakan proses melihat pengguna yang sudah terdaftar pada aplikasi pemesanan buket.
10.	Kelola Pesanan	Merupakan proses melihat pesanan yang masuk. Pesanan yang masuk nantinya akan diproses pada detail pesanan.
11.	Detail Pesanan	Merupakan proses menerima pesanan yang masuk. Pesanan nantinya akan diproses oleh admin dari mulai penjemputan barang pengiriman sampai pengiriman ke alamat pelanggan.
12.	Lacak Pengiriman	Merupakan proses melihat riwayat status saat proses pengiriman berlangsung.
13.	Invoice Pembelian	Merupakan proses melihat detail dokumen tagihan setiap pesanan berlangsung.
14.	Logout	Merupakan proses Ketika pengguna keluar dari sistem admin.

2. Diagram aktivitas

Diagram aktivitas adalah representasi visual yang menggambarkan alur kerja atau proses bisnis dalam sistem. Diagram ini memperlihatkan langkah-langkah aktivitas dan urutannya dalam proses pembayaran tunai pada transaksi e-commerce [5]. Diagram aktivitas admin yang ada pada Gambar 2 menunjukkan bahwa admin dapat mengakses sistem setelah berhasil melakukan login. Berbagai aktivitas yang dapat dilakukan oleh admin yaitu dapat melakukan pengelolaan produk beserta kategorinya. Admin juga dapat mengelola data banner,

data pesanan masuk yang dibuat oleh pelanggan, serta data pelanggan yang telah mendaftarkan akunnya pada aplikasi Android. Diagram aktivitas sistem informasi admin menunjukkan alur aktivitas ketika admin berhasil login, situs web akan menampilkan halaman dashboard.

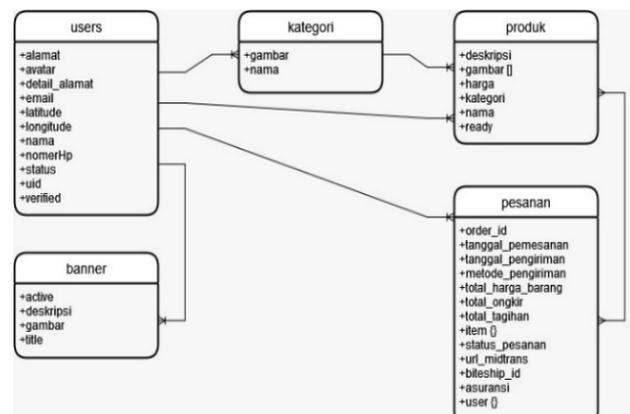


Gambar 2. Diagram aktivitas pada admin web

Aktivitas berikutnya admin dapat mengelola tabel data pada produk, kategori produk, *banner*, pengguna, dan pesanan. Selanjutnya sistem akan membaca data yang diakses dari basisdata dan menampilkan tabel data yang terbaru pada masing-masing halaman.

3. Diagram Model Data

Data yang dibutuhkan pada aplikasi sistem informasi admin yang akan dikembangkan dapat ditunjukkan dalam diagram model data pada Gambar 3



Gambar 3. Diagram Model Data



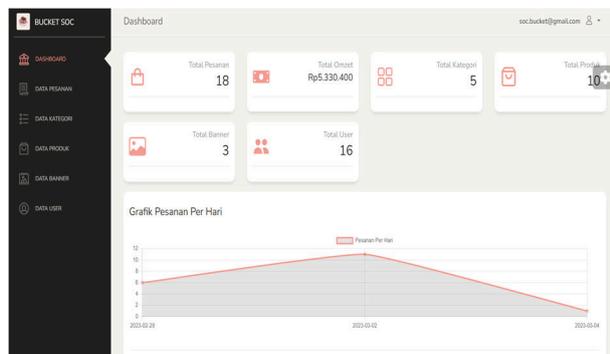
Basis data akan disimpan pada Firebase Realtime Database yang merupakan basis data NoSQL. NoSQL tidak seperti SQL yang memiliki ER dan diagram kelas [6]. Terlihat ada entitas *users*, kategori, produk, *banner*, dan pesanan. Entitas *users* memiliki relasi *one-to-many* terhadap entitas kategori, produk, *banner*, dan pesanan karena satu *users* dapat memiliki banyak pesanan. Entitas kategori memiliki relasi *one-to-many* terhadap entitas produk karena satu kategori dapat terdiri dari banyak produk. Entitas pesanan memiliki relasi *many-to-many* terhadap entitas produk karena banyak pesanan dapat terdiri banyak produk dan banyak produk yang dapat tersimpan pada banyak pesanan.

B. Implementasi Sistem

Setelah menyelesaikan tahap analisis kebutuhan dan tahap perancangan, Langkah berikutnya adalah implementasi. Implementasi didasarkan pada hasil perancangan yang sudah dibuat sebelumnya. Hasil implementasi Sistem Informasi Admin Berbasis Web pada Aplikasi Pemesanan Buket antara lain.

1. Tampilan Halaman Dashboard

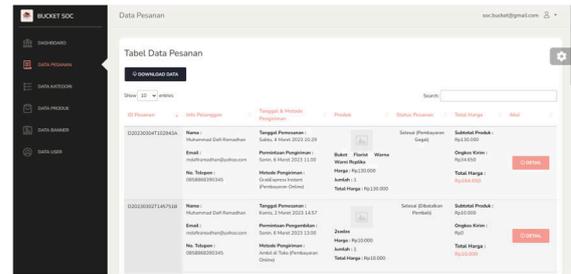
Pada Gambar 4 menampilkan halaman **Dashboard** pada aplikasi Sistem Informasi Admin "Bucket SOC". Halaman **Dashboard** berisi data yang menampilkan total pesanan yang dilakukan pelanggan, total omzet dari penjualan produk. Di halaman **Dashboard** juga ditampilkan data dari masing-masing halaman seperti **Total Kategori**, **Total Produk**, **Total Banner**, **Total User**, dan **Total Pesanan**.



Gambar 4. Halaman Dashboard

2. Tampilan Halaman Data Pesanan

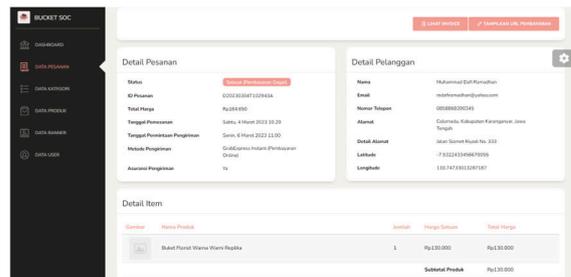
Pada halaman **Data Pesanan** yang ditunjukkan pada Gambar 5 menampilkan data dari pesanan yang masuk. Data ditampilkan berupa tabel yang berisi ID pesanan, info pelanggan, tanggal serta metode pengiriman, produk yang dipesan, **Status Pesanan**, dan **Total Harga** yang harus dibayar. Pada halaman pesanan ini juga terdapat tombol untuk melihat halaman **Detail Pesanan** sesuai ID pesanan



Gambar 5. Halaman Data Pesanan

3. Tampilan Halaman Detail Pesanan

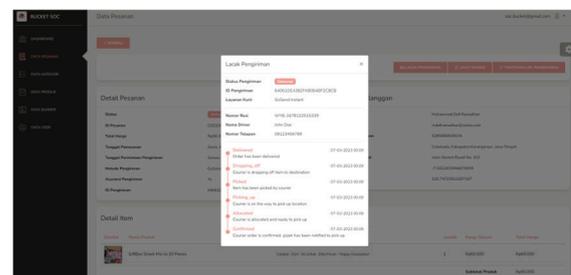
Tampilan halaman **Detail Pesanan** yang ditunjukkan pada Gambar 6 menampilkan detail pesanan sesuai ID pesanan. Halaman **Detail Pesanan** memiliki dua informasi tambahan yaitu **Detail Pelanggan** yang memesan dan **Detail Item** produk yang dibeli. Pada halaman **Detail Pesanan** admin melakukan konfirmasi pesanan dan memproses pesanan sesuai jadwal yang ditentukan. Setelah pesanan selesai admin dapat melakukan konfirmasi pesanan selesai. Admin dapat membatalkan pesanan jika tidak menyanggupi memproses pesanan.



Gambar 6. Halaman Detail Pesanan

4. Tampilan Modal Lacak Pengiriman

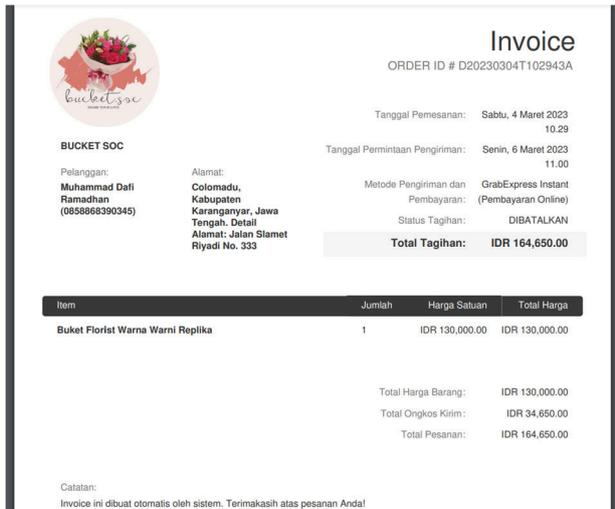
Fitur **Lacak Pengiriman** yang ditunjukkan pada Gambar 7 memungkinkan admin melihat kondisi atau status pesanan pada saat pengiriman. Terdapat **ID Pengiriman**, **Layanan Kurir**, **Nomor Resi** serta **Nama Driver**, dan **Nomor Telepon** pengemudi.



Gambar 7. Halaman Lacak Pengiriman

5. Tampilan Invoice Pesanan

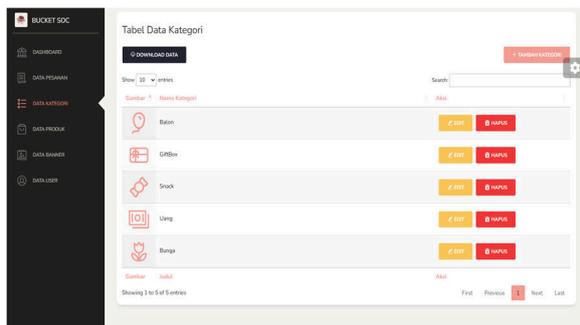
Pada fitur cetak tagihan yang ditunjukkan pada Gambar 8 admin dapat melihat detail tagihan pada pesanan yang dilakukan oleh pelanggan. Dokumen ini nantinya berguna untuk penjual dalam merekapitulasi atau melakukan perhitungan penghasilan dari penjualan produk.



Gambar 8. Halaman Invoice Pesanan

6. Tampilan Halaman Data Kategori

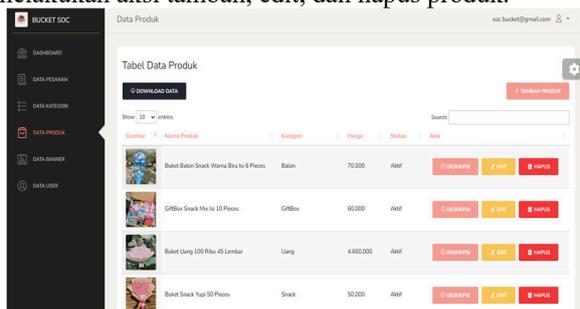
Pada gambar yang ditunjukkan pada Gambar 9 terlihat halaman **Data Kategori**. Halaman ini berisi informasi berkaitan data gambar dan nama kategori produk. Admin dapat menambahkan kategori ataupun melakukan edit kategori yang sudah ada. Pada halaman ini admin juga dapat menghapus kategori yang diinginkan.



Gambar 9. Halaman Data Kategori

7. Tampilan Halaman Data Produk

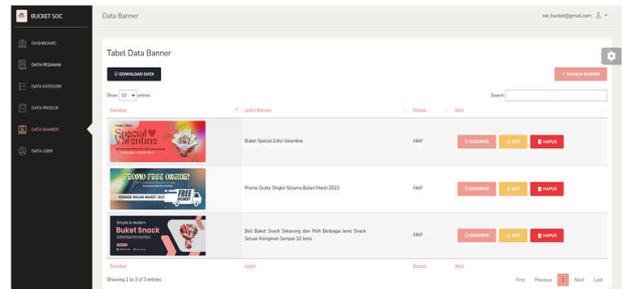
Pada halaman **Data Produk** admin dapat mengelola produk yang akan diposting nantinya ke aplikasi Android "Bucket SOC". Halaman produk menampilkan data produk seperti yang ditunjukkan pada Gambar 10 meliputi **Gambar**, **Nama**, **Kategori**, dan **Status** dari produk tertentu. Admin dapat melihat deskripsi, serta melakukan aksi tambah, edit, dan hapus produk.



Gambar 10. Halaman Data Produk

8. Tampilan Halaman Data Banner

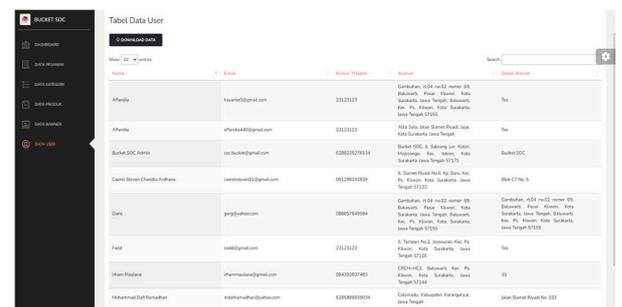
Pada halaman **Data Banner** yang ditunjukkan Gambar 11 menampilkan daftar banner yang ada berupa gambar, nama, dan status *banner*. Pada halaman ini admin dapat mengelola seperti tambah, edit, dan hapus *banner*. Kemudian pada halaman **Data Banner** admin dapat melihat deskripsi *banner*.



Gambar 11. Halaman Data Banner

9. Tampilan Halaman Data User

Halaman Data User yang ditunjukkan Gambar 12 menampilkan data user baik admin atau pembeli. Pada halaman ini data yang ditampilkan adalah nama, email, nomor telepon, serta alamat, dan detail alamat dari pengguna yang terdaftar.



Gambar 12. Halaman Data User

10. Tampilan Halaman Edit Profil

Pada halaman **Edit Profile** yang ditunjukkan Gambar 13 admin dapat merubah informasi mengenai akun miliknya seperti foto profil, nama, nomor telepon, alamat, detail alamat, lokasi lintang, dan bujur. Data tersebut jika diubah akan mempengaruhi titik awal lokasi pengiriman produk dan juga lokasi toko penjual yang ditampilkan pada aplikasi Android. Untuk data email tidak dapat diubah. Sedangkan *password* dapat diubah dengan mengklik pada tombol **Reset Password** yang akan mengirimkan tautan reset *password* ke email Admin tersebut

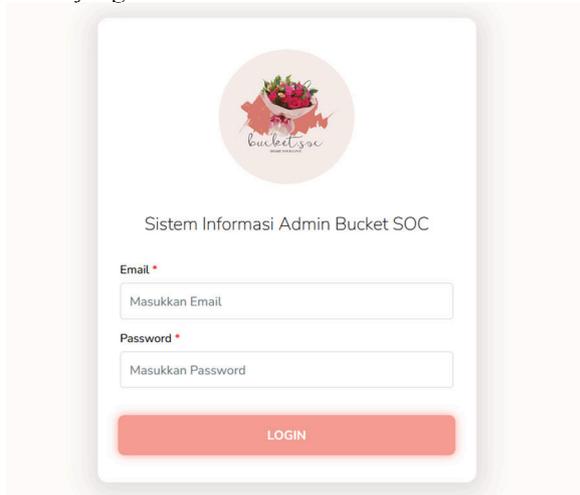


Gambar 13. Halaman Edit Profil



11. Tampilan Halaman Login

Halaman **Login** yang ditunjukkan pada Gambar 14 menampilkan formulir untuk melakukan login. Data yang harus diisi yaitu *email* dan *password* dari akun Admin yang telah terdaftar.



Gambar 14. Halaman Login

C. Pengujian Sistem

1. Pengujian Kotak-Hitam

Pengujian kotak-hitam adalah pengujian yang dilakukan untuk mengamati hasil masukan dan keluaran dari perangkat lunak tanpa mengetahui struktur kode dari perangkat lunak tersebut. Pengujian ini dilakukan di akhir pembuatan perangkat lunak untuk mengetahui apakah perangkat lunak dapat berfungsi dengan baik [7].

Hasil pengujian kotak-hitam dilakukan pada aplikasi sistem informasi admin berbasis web dapat dilihat pada Tabel 2

Tabel 2. Hasil Pengujian Kotak Hitam

No	Skenario	Hasil Pengujian
1	Admin dapat melakukan <i>login</i> dengan akun yang telah didaftarkan sebelumnya	Berhasil
2	Admin dapat melihat dan melakukan perubahan data akun miliknya pada halaman Edit Profile	Berhasil
3	Admin dapat melakukan <i>reset password</i> jika lupa terhadap <i>password</i> akun miliknya	Berhasil
4	Admin dapat melakukan <i>logout</i> untuk keluar dari akun miliknya	Berhasil
5	Admin dapat melihat berbagai statistik data dan grafik pada halaman Dashboard	Berhasil
6	Admin dapat menambahkan, melakukan edit, dan menghapus data kategori produk pada halaman Data Kategori	Berhasil
7	Admin dapat menambahkan, melakukan edit, melihat detail deskripsi, dan menghapus data produk pada halaman Data Produk	Berhasil

8	Admin dapat menambahkan, melakukan edit, melihat detail deskripsi, dan menghapus data banner pada halaman Data Banner	Berhasil
9	Admin dapat melihat informasi <i>user</i> yang terdaftar pada halaman Data User	Berhasil
10	Admin dapat melakukan penyortiran data dan mengatur jumlah data yang tampil pada setiap halaman tabel yang ada di halaman Data Pesanan, Data Kategori, Data Produk, Data Banner, dan Data User	Berhasil
11	Admin dapat mengunduh data pada tabel yang ada di halaman Data Pesanan, Data Kategori, Data Produk, Data Banner, dan Data User	Berhasil
12	Admin dapat melakukan pencarian sesuai kata kunci yang diinginkan pada halaman Data Pesanan, Data Kategori, Data Produk, Data Banner, dan Data User	Berhasil
13	Admin dapat melihat data pesanan yang masuk pada halaman Data Pesanan	Berhasil
14	Admin dapat melakukan konfirmasi pesanan yang masuk, memproses penjemputan barang untuk dikirim, melihat tagihan pembayaran, mengganti jadwal pengiriman, melihat status pengiriman, dan membatalkan pesanan pada halaman Detail Pesanan	Berhasil

Berdasarkan pengujian kotak hitam yang telah dilakukan terhadap 14 skenario pada aplikasi sistem informasi admin berbasis web, diperoleh hasil bahwa seluruh skenario tersebut berhasil dijalankan

V. PENUTUP

A. Kesimpulan

Semua tahapan yang dilakukan dalam penelitian ini, mulai dari analisis, perancangan, pengimplementasian sampai pengujian dapat disimpulkan beberapa hal sebagai berikut: sistem informasi admin berbasis web pada aplikasi pemesanan buket dibuat dengan menggunakan pustaka ReactJS dan metode *Waterfall*. Data dari Midtrans API dan Biteship API dapat disimpan dan diakses untuk ditampilkan pada aplikasi dengan menggunakan metode GET dan POST. Hasil pengujian kotak-hitam menunjukkan bahwa aplikasi berfungsi dengan baik. Sehingga penjual dalam mengelola produk yang dijual dan memproses pesanan dapat dilakukan secara daring.

B. Saran

Salah satu saran untuk penelitian selanjutnya adalah melakukan pengembangan sistem yang serupa dengan menggunakan kerangka kerja atau pustaka yang lain selain ReactJS. Hal ini dapat memberikan perbandingan kinerja dan kelebihan dari masing-masing kerangka kerja



atau pustaka yang digunakan. Beberapa contoh kerangka kerja atau pustaka yang dapat digunakan adalah VueJS atau AngularJS. Selain itu, saran lainnya adalah melakukan penelitian lebih lanjut dengan mengembangkan aplikasi serupa untuk bidang usaha yang berbeda, misalnya: sistem informasi admin pada aplikasi pemesanan kartu ucapan. Aplikasi ini dapat menggunakan metode yang sama dengan penelitian aplikasi sistem informasi admin Bucket SOC, yaitu ReactJS dan metode *Waterfall*.

DAFTAR PUSTAKA

- [1] A. M. S. Huda dan Y. Fernando, "E-Ticketing Penjualan Tiket Event Musik Di Wilayah Lampung Pada Karcismu Menggunakan Library Reactjs," *Jurnal Teknologi Dan Sistem Informasi*, vol. 2, no. 1, hlm. 96–103, 2021.
- [2] J. Panjaitan dan A. F. Pakpahan, "Perancangan Sistem E-Reporting Menggunakan ReactJS dan Firebase," *Jurnal Teknik Informatika Dan Sistem Informasi*, vol. 7, no. 1, 2021.
- [3] A. A. Wahid, "Analisis Metode Waterfall Untuk Pengembangan Sistem Informasi," *J. Ilmu-ilmu Inform. dan Manaj. STMIK*, no. November, hlm. 1–5, 2020.
- [4] F. Fauzan, R. Reza, et al., "Use case diagram similarity measurement: A new approach," dalam *2019 12th International Conference on Information & Communication Technology dan System (ICTS)*, IEEE, 2019.
- [5] M. Syarif, M. Muhamad, dan W. Nugraha, "Pemodelan diagram UML sistem pembayaran tunai pada transaksi e-commerce," *JTIK (Jurnal Teknik Informatika Kaputama)*, vol. 4, no. 1, pp. 64–70, 2020.
- [6] Talha Awan, "How to Draw NoSQL Data Model Diagram?," <https://www.techighness.com/post/how-to-draw-no-sql-data-model-diagram/>, 16 Mei 2022, [diakses pada 5 April 2023].
- [7] H. Nurfauziah dan I. Jamaliyah, "Perbandingan Metode Testing Antara Blackbox Dengan Whitebox Pada Sebuah Sistem Informasi," *Jurnal Visualika STMIK Muhammadiyah Jakarta*, vol. 8, no. 2, hlm. 105–113, 2022.



©2023. This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/).