



Implementasi Flutter Dalam Pengembangan Aplikasi Pemesanan Sangkar Burung Berbasis Android

Andri Kurnia Maulana¹, Nizirwan Anwar², Binastya Anggara Sekti³

^{1,2}Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Esa Unggul

³Program Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Esa Unggul

andri.kurnia.maulana@gmail.com¹, nizirwan.anwar@esaunggul.ac.id², anggara@esaunggul.ac.id³

Abstract

In recent years, mobile technology, including the field of electronic commerce, has seen rapid growth. Bird cages have become one of the most sought-after products in the bird trading industry. However, there are several issues to consider during the ordering process, which often consume time and effort. The objective of this research is to create and develop an Android-based bird cage ordering application with a user-friendly interface that allows people to quickly and easily select and purchase bird cages. The research methodology involves identifying the requirements for aspects such as material, size, bird cage design, and shipping information. Flutter enables user interface designers to display a list of bird cages with photos and details, as well as customization options. Firebase provides backend storage for storing history, authentication, and orders. The application features include registration, login, searching for bird cages, viewing details with images, ordering, tracking orders, and logging out. By offering a variety of bird cage options, this application streamlines the bird cage ordering process. Customers can monitor the status of their orders and easily choose the most suitable one. Flutter simplifies cross-platform development. Using the Flutter framework and Firebase as the backend, users can select and order bird cages according to customer or application user needs.

Keywords: Flutter, Dart, Firebase, Android Mobile, Booking

Abstrak

Dalam beberapa tahun terakhir, teknologi mobile, termasuk bidang perdagangan elektronik, telah berkembang pesat. Sangkar burung telah menjadi salah satu produk yang paling dicari dalam industri perdagangan burung. Namun, ada beberapa masalah yang perlu diperhatikan selama proses pemesanan, yang sering memakan waktu dan tenaga. Tujuan dari penelitian ini adalah untuk membuat dan mengembangkan sebuah aplikasi pemesanan sangkar burung berbasis Android dengan antarmuka pengguna yang ramah pengguna yang memungkinkan orang memilih dan membeli sangkar burung dengan cepat dan mudah. Metode penelitian dilakukan dengan identifikasi kebutuhan terhadap aspek bahan, ukuran, desain sangkar burung, dan informasi pengiriman. Flutter memungkinkan desainer antarmuka pengguna menampilkan daftar sangkar dengan foto dan detail serta opsi penyesuaian. Firebase memungkinkan penyimpanan backend untuk menyimpan histori, otentikasi, dan pesanan. Aplikasi memiliki fitur seperti pendaftaran, masuk, mencari sangkar, melihat detail dengan gambar, memesan, melacak pesanan, dan logout. Dengan melihat berbagai pilihan sangkar burung, aplikasi ini membantu proses pemesanan sangkar burung. Pelanggan dapat memantau status pemesanan mereka dan dengan mudah memilih yang paling sesuai. Flutter mempermudah pengembangan lintas platform. Menggunakan framework Flutter dan Firebase sebagai backend, maka pengguna dapat memilih, dan memesan sangkar burung sesuai kebutuhan pelanggan atau pengguna aplikasi.

Kata kunci: Flutter, Dart, Firebase, Android Mobile, Booking

1. Pendahuluan

Kemampuan untuk meningkatkan perekonomian menjadi masalah penting dalam memenuhi kebutuhan sehari-hari. Banyak instansi, pemerintah, dan lembaga melakukan pengujian kondisi perekonomian masyarakat karena banyak aktivitas yang membutuhkan hal tersebut. Kondisi ini adalah masalah pada mata pencaharian yang menghambat kemampuan seseorang untuk memenuhi kesehariannya dikarenakan banyaknya penjualan kerajinan tangan seperti sangkar

burung di Indonesia. Industri sangkar burung berasal dari Jawa Tengah memiliki potensi pengembangan yang besar, bahkan telah dikenal di Kabupaten Tangerang[1]. Berdasarkan data dari Badan Pusat Statistik (BPS) tahun 2020, terdapat 41.023 unit perusahaan mikro atau kecil di Jawa Barat yang bergerak dalam bidang kerajinan berbahan kayu atau bambu, termasuk 34 unit perusahaan yang memproduksi kerajinan tersebut, namun tidak termasuk dalam kategori furnitur. [2].

Masih diperlukan waktu dan perjalanan yang cukup jauh untuk mencari pelanggan yang berminat membeli sangkar burung. Proses ini dapat memakan banyak waktu dan biaya, baik untuk mencari pelanggan yang tepat maupun melakukan perjalanan ke berbagai daerah. Sebab, para penjual harus berkeliling dan mencari pelanggan yang ingin membeli sangkar burung, yang memerlukan tenaga dan bensin yang cukup besar. [3].

Dalam sebuah jurnal penelitian lain, disebutkan bahwa Presiden Joko Widodo pernah mengatakan bahwa hobi memelihara burung di Indonesia berhasil membangkitkan perekonomian rakyat, melalui penangkaran, pakan, sangkar, dan obat-obatan. Jurnal tersebut berjudul "PENGEMBANGAN KERAJINAN SANGKAR BURUNG KELURAHAN KADIPIRO, SURAKARTA, JAWA TENGAH". [4].

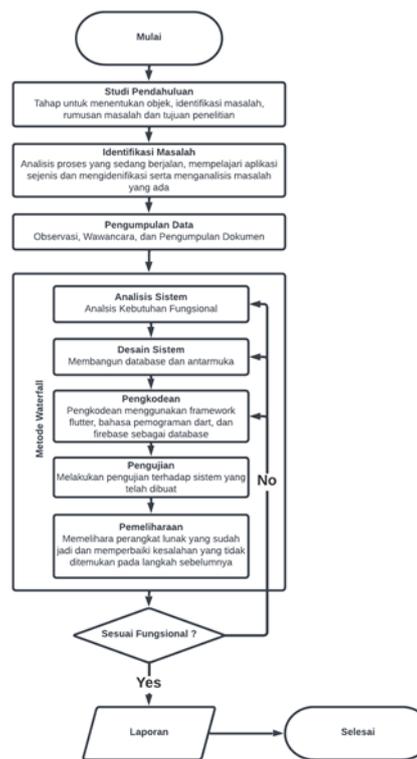
Dari masalah dan penjelasan di atas, solusi untuk masalah tersebut adalah dengan membuat aplikasi berbasis Android yang dapat membantu dalam proses pemesanan. Aplikasi ini merupakan aplikasi pemesanan sangkar burung yang memungkinkan pengguna untuk memesan sangkar burung sesuai dengan preferensinya. Melalui aplikasi ini, pelanggan dapat memesan sangkar burung secara langsung dari pembuat sangkar burung, yang kemudian akan dikirimkan langsung ke pintu pelanggan berdasarkan pesanan yang dipesan oleh pelanggan. Selain itu, pembuat sangkar burung dapat menambahkan menu dengan menentukan kategori sesuai dengan kebutuhan pelanggan. [5].

Penelitian ini berfokus pada bagaimana meningkatkan efisiensi dan efektivitas pemesanan sangkar burung berbasis android menggunakan teknologi Flutter dengan bahasa dart sebagai Bahasa pemrogramannya. Dengan aplikasi mobile yang terintegrasi dengan layanan backend basis data Firebase biaya dan waktu yang diperlukan dapat diminimalkan biaya sehingga dapat melakukan penjualan sangkar burung dengan cepat dan dalam skala kuantitas yang lebih besar.

2. Metode Penelitian

2.1. Tahapan Penelitian

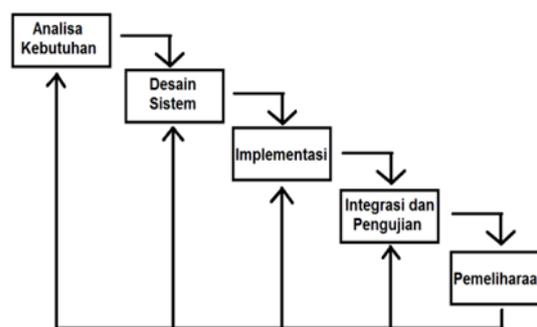
Bagian ini akan dijelaskan langkah-langkah yang dilakukan untuk pembangunan aplikasi pemesanan sangkar burung berbasis Android. Ada beberapa tahapan pada flowchart penelitian ini, yaitu studi pendahuluan, studi literatur, pengumpulan data, analisis sistem, pembuatan desain sistem, pengkodean, pengujian sistem, dan pembuatan Tugas Akhir. Flowchart penelitian ini dapat dilihat pada Gambar 1 sebagai berikut :



Gambar 1 Flowchart Penelitian

2.2. Metode Pengembangan Sistem

Metode pembuatan sistem menggunakan pendekatan Waterfall untuk membuat aplikasi sangkar burung. Model Waterfall menurut Syafnidawaty, 2020 [6] adalah metode yang menggambarkan pendekatan sebagai sebuah model pengembangan sekuensial yang bersifat sistematis dan berurutan saat membangun perangkat lunak. Tahapan-tahapan pembuatan aplikasi dengan metode Waterfall ini digambarkan dalam alur diagram pada gambar 2 sebagai berikut :



Gambar 2 Model Pengembangan Waterfall

Berikut adalah penjelasan dari tahap-tahap yang akan dilakukan berdasarkan model Waterfall ini menurut :

Analisis Kebutuhan

Tahap ini pengembang sistem diperlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut. Informasi ini biasanya dapat diperoleh melalui

wawancara, observasi, diskusi atau survei langsung. Informasi dianalisis untuk mendapatkan data yang dibutuhkan oleh pengguna.

Desain Sistem

Spesifikasi kebutuhan dari tahap sebelumnya akan dipelajari dalam fase ini dan desain sistem disiapkan. Desain Sistem membantu dalam menentukan perangkat keras (hardware) dan sistem persyaratan dan juga membantu dalam mendefinisikan arsitektur sistem secara keseluruhan.

Implementasi

Pada tahap ini, sistem pertama kali dikembangkan di program kecil yang disebut unit, yang terintegrasi dalam tahap selanjutnya. Setiap unit dikembangkan dan diuji untuk fungsionalitas yang disebut sebagai unit testing.

Integrasi dan Pengujian

Seluruh unit yang dikembangkan dalam tahap implementasi diintegrasikan ke dalam sistem setelah pengujian yang dilakukan masing-masing unit. Setelah integrasi seluruh sistem diuji untuk mengecek setiap kegagalan maupun kesalahan.

Pemeliharaan

Tahap akhir dalam model Waterfall. Perangkat lunak yang sudah jadi, dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya. Perbaikan implementasi unit sistem dan peningkatan jasa sistem sebagai kebutuhan baru.

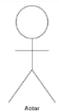
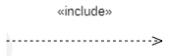
2.4. Desain Pemodelan Sistem UML

Unified Modeling Language (UML) adalah salah satu bahasa dasar yang banyak digunakan dalam dunia industri guna mendefinisikan kebutuhan, membuat desain dan analisis, dan menggambarkan arsitektur dalam pemrograman yang berorientasikan pada objek (Rosa. & Shalahuddin., 2019). [7].

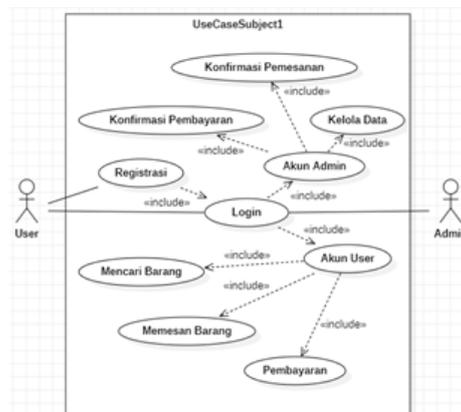
Use Case Diagram

Menurut (P. Danenas et al., 2020) [8] Use case atau diagram use case merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. Syarat penamaan pada use case adalah nama didefinisikan sesimpel mungkin dan dapat dipahami. Berikut ini adalah tabel penjelasan serta keterangan yang akan dipergunakan dalam pembuatan use case diagram.

Tabel 1 Daftar Simbol Use Case Diagram

Simbol	Keterangan
	<i>Actor</i> : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan use case.
	<i>Use Case</i> : Abstraksi dan interaksi antara sistem dan aktor.
	<i>Association</i> : Abstraksi dari penghubung antara aktor dengan use case.
	<i>Generalisasi</i> : Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan use case.
	Menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lainnya.
	Menunjukkan bahwa suatu use case merupakan tambahan fungsional dari use case lainnya jika suatu kondisi terpenuhi.

Dari penjelasan tabel diatas dapat dilakukan perancangan use case diagram aplikasi dibawah ini memaparkan hak dan kemampuan aktor dalam berinteraksi dengan sistem aplikasi lewat fitur-fitur yang tersedia dalam aplikasi pemesanan sangkar burung ini:

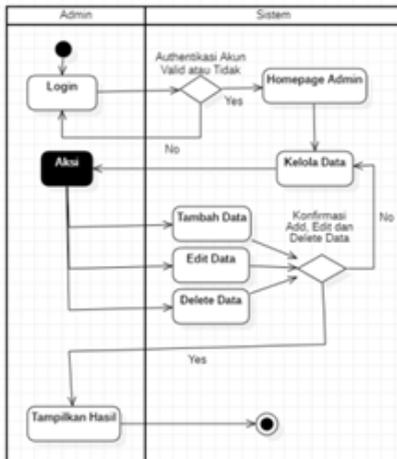


Gambar 3 Use Case Diagram

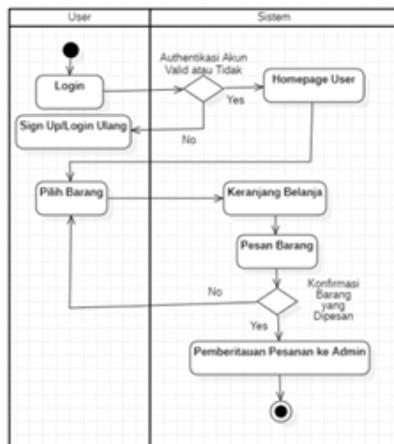
2. Activity Diagram

Diagram aktivitas atau activity diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak [9]. Berikut ini adalah contoh

gambar dari activity diagram admin dan user yang akan digunakan dalam perancangan aplikasi pemesanan sangkar burung.



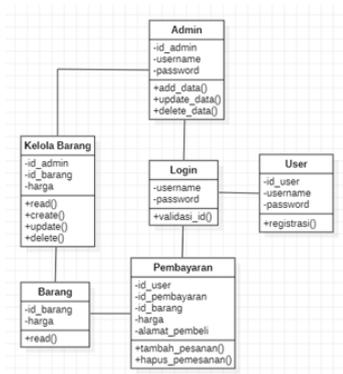
Gambar 4 Activity Diagram Admin



Gambar 5 Activity Diagram User

3. Class Diagram

Menurut (M. Thomas et al., 2021) [10] diagram kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelaskelas yang akan dibuat untuk membangun sistem. Berikut ini adalah contoh gambar dari activity diagram yang akan digunakan dalam perancangan aplikasi pemesanan sangkar burung.

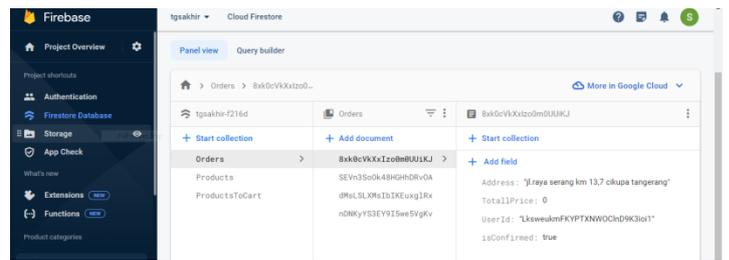


Gambar 6 Class Diagram

3. Hasil dan Pembahasan

3.1. Desain Aplikasi

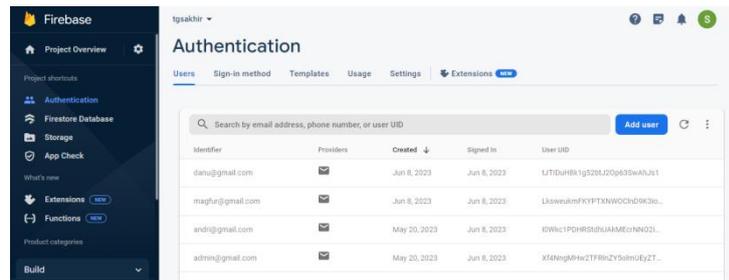
1. Tampilan Firestore Database



Gambar 7 Database Firestore

Gambar diatas adalah tampilan database yang berfungsi sebagai database atau penyimpanan data dari aplikasi yang digunakan untuk menampilkan data pada aplikasi.

2. Tampilan Database User dan Admin

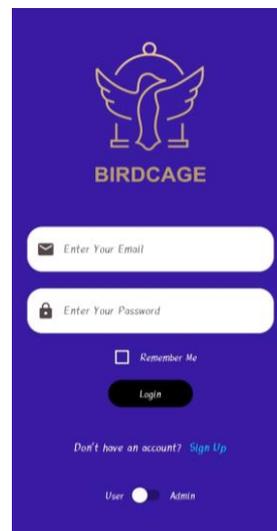


Gambar 8 Database User dan Admin

Gambar diatas adalah tampilan authentication untuk admin dan user agar dapat login pada aplikasi.

3. Tampilan Menu Login

Gambar 9 dibawah ini adalah hasil tampilan Menu Login.

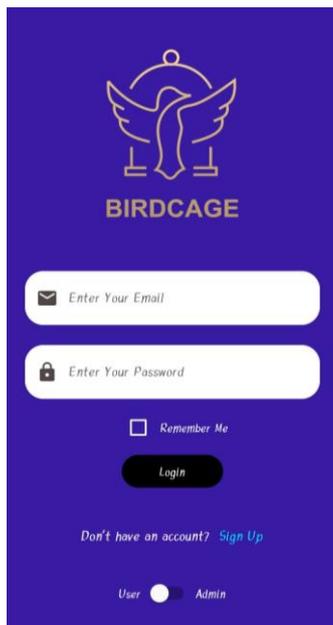


Gambar 9 Menu Login

Gambar diatas adalah tampilan dari aplikasi untuk melakukan login aplikasi dengan tombol switch untuk berubah cara login untuk admin dan user.

4. Tampilan Homepage Admin

Gambar 10 dibawah ini adalah tampilan Homepage Admin.



Gambar 10 Homepage Admin

Gambar diatas adalah tampilan dari beranda admin yang memiliki fungsi untuk menambah, mengubah, menghapus, dan melihat konfirmasi pemesanan.

5. Tampilan Add Product Admin

Gambar 11 dibawah ini adalah hasil tampilan Add Product Admin.

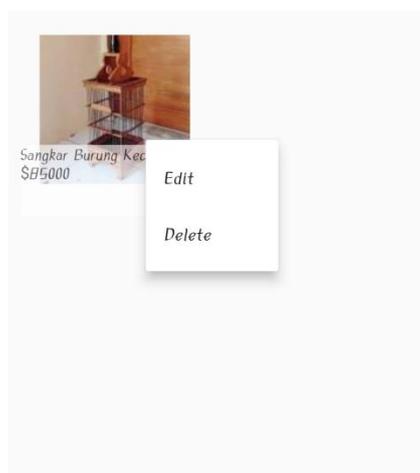


Gambar 11 Add Product Admin

Gambar diatas adalah tampilan admin untuk menambah produk yang akan dijual dalam aplikasi pemesanan barang.

6. Tampilan Delete Product Admin

Gambar 12 dibawah ini adalah hasil tampilan Delete Product Admin.



Gambar 12 Delete Product Admin

Gambar diatas adalah tampilan untuk menghapus atau mengubah data-data dari produk yang ada pada aplikasi tersebut.

7. Tampilan Edit Product Admin

Gambar 13 dibawah ini adalah tampilan Edit Product Admin.

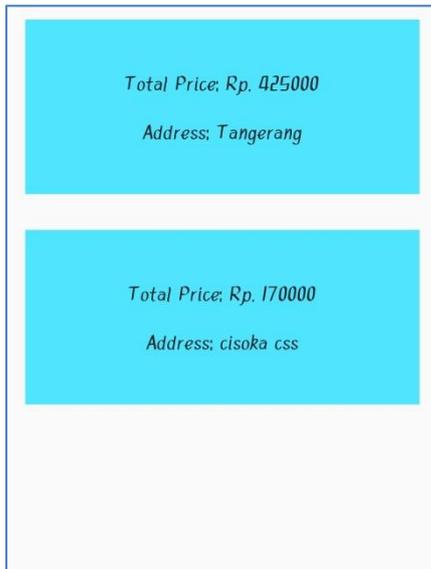


Gambar 13 Edit Product Admin

Gambar diatas tampilan fitur admin yang berfungsi untuk mengubah data-data produk yang dijual dalam aplikasi pemesanan tersebut.

8. Tampilan View Orders Admin

Gambar 14 dibawah ini adalah tampilan View Orders Admin.

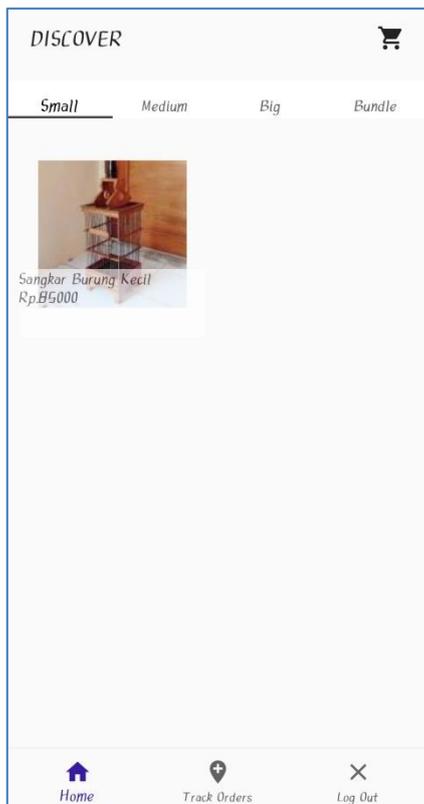


Gambar 14 View Orders

Gambar diatas adalah tampilan dari aplikasi untuk mengkonfirmasi pemesanan yang dipesan oleh pengguna aplikasi tersebut.

9. Tampilan Homepage User

Gambar 15 dibawah ini adalah tampilan Homepage User.

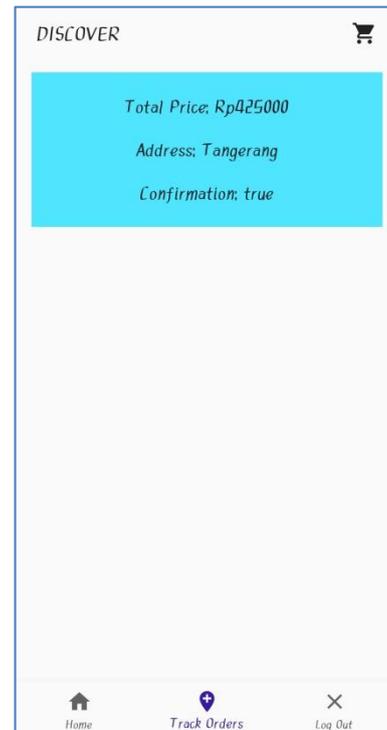


Gambar 15 Homepage User

Gambar diatas adalah tampilan dari beranda user/pembeli yang menggunakan aplikasi pemesanan tersebut.

10. Tampilan Track Orders User

Gambar 16 tampilan Track Orders User.

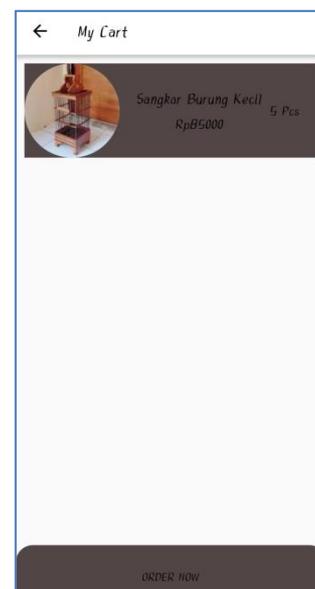


Gambar 16 Track Orders User

Gambar diatas adalah tampilan dari pemesanan barang yang dipesan user/pembeli telah dikonfirmasi oleh penjual barang.

11. Tampilan Cart User

Gambar 17 dibawah ini adalah tampilan Cart User.



Gambar 17 Cart User

Gambar diatas adalah tampilan dari keranjang pemesanan yang dipesan oleh pengguna aplikasi atau user dalam melakukan pemesanan barang.

3.2. Hasil Pengujian (Blackbox Testing)

1. Hasil Pengujian Admin

Tabel 2 Blackbox Testing Admin

No	Skenario Pengujian	Test	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Login	Semua data tidak diisi	Error muncul warning yang memerintahkan isi data dengan benar	Sesuai Harapan	Valid
2	Admin add dan edit	Semua data tidak diisi	Error pada kolom yang tidak diisi	Sesuai Harapan	Valid
3	Product lalu add atau edit	Semua data diisi lengkap	Tetap dalam form add atau edit tanpa muncul	Sesuai Harapan	Valid
4	Admin add dan edit	Tidak dikonfirmasi atau ditolak	Tidak dapat melakukan proses pengiriman barang	Sesuai Harapan	Valid

2. Hasil Pengujian User

Tabel 3 Blackbox Testing User

No	Skenario Pengujian	Test	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1	Login	Semua data tidak diisi	Error muncul warning yang memerintahkan isi data dengan benar	Sesuai Harapan	Valid
2	User melakukan add product ke cart	Data produk kosong	Tidak dapat melakukan penam	Sesuai Harapan	Valid

			bahan data ke keranjang pemesanan		
3	User melakukan cancel order	Cancel order	Dapat melakukan cancel order baik sudah dikonfirmasi atau belum dikonfirmasi	Sesuai Harapan	Valid

4. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan bahwasanya aplikasi yang telah dibangun dapat meningkatkan keefisienan pemesanan sangkar burung dibandingkan dengan melakukan pemesanan secara manual yang memiliki kekurangan sebagai berikut memakan waktu yang lama karena pemesanan yang terlalu mendadak, serta adanya kesalahan catat disetiap interkasinya. Serta aplikasi ini juga memberikan informasi seputar sangkar burung.

Adapun kelebihan maupun kekurangan aplikasi ini, saya selaku peneliti berikut saya berikan beberapa saran yang dapat digunakan untuk mengembangkan aplikasi ini lebih lanjut lagi :

Diharapkan pada fitur pemesanan dikembangkan lebih lanjut lagi, dimana aplikasi bisa dijadikan sebagai point of sale.

Daftar Rujukan

- [1] T. M. Squires et al., "Citizen science rapidly delivers extensive distribution data for birds in a key tropical biodiversity area," *Glob. Ecol. Conserv.*, vol. 28, hal. e01680, 2021, doi: <https://doi.org/10.1016/j.gecco.2021.e01680>.
- [2] Badan Pusat Statistik (BPS), "Data Jumlah Perusahaan Kabupaten Tangerang Di Bagian Industri besar dan sedang 2015-2018," 2018. <https://tangerangkab.bps.go.id/statictable/2015/05/16/30/jumlah-perusahaan-dan-tenaga-kerja-dari-industri-besar-dan-sedang-menurut-gol-industri-2013.html>
- [3] R. Setiawan dan W. Nugroho, "Rancang Bangun Aplikasi Pemesanan Makanan Berbasis Android," *Journal of Information Systems and Informatics*, vol. 3, no. 2. Asosiasi Perguruan Tinggi Informatika dan Komputer (APTIKOM) Sumsel, hal. 329-340, 2021. doi: 10.33557/journalisi.v3i2.132.
- [4] Budi Setiastuti, "Abdi Seni," vol. 5, no. 1, hal. 68-74, 2011, [Daring]. Tersedia pada: <https://jurnal.isi-ska.ac.id/index.php/abdiseni/article/view/3041/2568>
- [5] A. M. Suzana, "Analisis Dan Perancangan Aplikasi Pemesanan Tiket Bus Berbasis Android," *Jurnal Informatika dan Rekayasa Perangkat Lunak*, vol. 2, no. 3. Universitas Teknokrat Indonesia, hal. 353-360, 2021. doi: 10.33365/jatika.v2i3.1235.
- [6] Syafnidawaty, "Metode Pendekatan Waterfall," 04 April 2020, 2020. <https://raharja.ac.id/2020/04/04/metode-waterfall/>

- [7] Rosa. dan Shalahuddin., *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek Edisi Revisi*. ISBN : 9786021514054. Bandung, 2019.
- [8] P. Danenas, T. Skersys, dan R. Butleris, "Natural language processing-enhanced extraction of SBVR business vocabularies and business rules from UML use case diagrams," *Data Knowl. Eng.*, vol. 128, hal. 101822, 2020, doi: <https://doi.org/10.1016/j.datak.2020.101822>.
- [9] M. Abbas, R. Rioboo, C.-B. Ben-Yelles, dan C. F. Snook, "Formal modeling and verification of UML Activity Diagrams (UAD) with FoCaLiZe," *J. Syst. Archit.*, vol. 114, hal. 101911, 2021, doi: <https://doi.org/10.1016/j.sysarc.2020.101911>.
- [10] M. Thomas, I. Mihaela, R. M. Andrianjaka, D. W. Germain, dan I. Sorin, "Metamodel based approach to generate user interface mockup from UML class diagram," *Procedia Comput. Sci.*, vol. 184, hal. 779–784, 2021, doi: <https://doi.org/10.1016/j.procs.2021.03.096>.