

**PENGEMBANGAN SISTEM INFORMASI PENJUALAN IKAN  
CUPANG BERBASIS WEB DI LABETTA SOLO**



**Disusun sebagai salah satu syarat memperoleh Gelar Strata I  
pada Program Studi informatika Fakultas Komunikasi dan Informatika**

**Oleh:  
DANDUNG RAHMATDHAN  
L200170098**

**PROGRAM STUDI INFORMATIKA  
FAKULTAS KOMUNIKASI DAN INFORMATIKA  
UNIVERSITAS MUHAMMADIYAH SURAKARTA  
2021**

**HALAMAN PERSETUJUAN**

**PENGEMBANGAN SISTEM INFORMASI PENJUALAN IKAN CUPANG  
BERBASIS WEB DI LABETTA SOLO**

**PUBLIKASI ILMIAH**

oleh:

**DANDUNG RAHMATDHAN**  
**L200170098**

Telah diperiksa dan disetujui untuk diuji oleh:  
Dosen Pembimbing



**Dedi Gunawan, S.T., M.Sc., Ph.D**

**NIK.1305**

**HALAMAN PENGESAHAN**

**PENGEMBANGAN SISTEM INFORMASI PENJUALAN IKAN  
CUPANG BERBASIS WEB DI LABETTA SOLO**

**OLEH**

**DANDUNG RAHMATDHAN**

**L200170098**

**Telah dipertahankan di depan Dewan Penguji  
Fakultas Komunikasi dan Informatika  
Universitas Muhammadiyah Surakarta  
Pada hari Kamis, 15 Juli 2021  
dan dinyatakan telah memenuhi syarat**

**Dewan Penguji:**

**1. Dedi Gunawan, S.T, M.Sc, Ph.D.**

**(Ketua Dewan Penguji)**

**2. Dr. Endah Sudarmilah, S.T, M.Eng**

**(Anggota I Dewan Penguji)**

**3. Yusuf Sulisty Nugroho, S.T, M.Eng**

**(Anggota II Dewan Penguji)**



**Dekan  
Fakultas Komunikasi dan Informatika**



**Nurgiyatna, S.T., M.Sc., Ph.D.  
NIK. 881**

## **PERNYATAAN**

Dengan ini saya menyatakan bahwa dalam publikasi ilmiah ini tidak terdapat karya yang pernah diajukan untuk memperoleh gelar kesarjanaan di suatu perguruan tinggi dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan orang lain, kecuali secara tertulis diacu dalam naskah dan disebutkan dalam daftar pustaka.

Apabila kelak terbukti ada ketidakbenaran dalam pernyataan saya di atas, maka akan saya pertanggungjawabkan sepenuhnya.

**Surakarta, 29 Juli 2021**

Penulis



**DANDUNG RAHMATDHAN**

**L200170098**

# PENGEMBANGAN SISTEM INFORMASI PENJUALAN IKAN CUPANG BERBASIS WEB DI LABETTA SOLO

## Abstrak

Labetta Solo merupakan penjual ikan cupang hias yang berlokasi di belakang ayam goreng widuran, Kepatihan Kulon, Kec. Jebres, Kota Surakarta. Proses penjualan dan transaksi yang dilakukan masih menggunakan model transaksi manual yaitu *Customer* harus ke toko untuk melihat langsung koleksi ikan cupang untuk melakukan transaksi, atau dengan cara melihat koleksi ikan cupang di media *social* labetta solo untuk kemudian melakukan COD (*Cash On Delivery*) dengan penjual untuk transaksi, dengan pembuatan sistem ini nantinya labetta solo tetap dapat berjualan dan *Customer* tetap dapat membeli ikan cupang dengan mudah. Sistem ini nantinya akan dibuat menggunakan metode Waterfall meliputi analysis, design, coding, testing, implementation sistem ini nantinya akan dibuat menggunakan database MySQL dan PHP *CodeIgniter* sebagai bahasa pemrogramannya. Hasil yang diharapkan nantinya penjual dapat menjual ikan melalui system ini dan terdapat rating atau grade dari kualitas ikan sehingga memudahkan *Customer* dalam melakukan pemilihan ikan yang akan di beli, system ini juga terdapat lokasi dari Labetta Solo untuk memudahkan bagi *Customer* yang ingin berkunjung langsung ke lokasi dan untuk mempermudah komunikasi kepada *Customer* akan tersedia link menuju WhatsApp dari penjual. Berdasarkan hasil pengujian dengan *black-box* dan SUS testing menunjukkan bahwa sistem tidak memiliki error, selain itu uji penerimaan *user* terhadap aplikasi juga tergolong tinggi dengan prosentase sekitar 76%.

**Kata Kunci:** ikan cupang, web, Waterfall, PHP

## Abstract

Labetta Solo is a seller of ornamental betta fish located at Kepatihan Kulon, Kec. Jebres, Surakarta City, Central Java. The process of sales and transactions is still using the manual transaction model, where *Customers* have to go to the store to see the Betta fish collection to make transactions. In some cases, *Customers* are visiting the Betta fish collection on Labetta Solo *social* media and doing COD (Cash on Delivery) with the seller for transactions. However, such activities should be limited specifically meeting with other people due to COVID-19 pandemic. Therefore, to address the challenge and increase the sales we design a web-based application so that the Labetta Solo still be able to perform transaction and *Customers* can still buy Betta fish easily. The system is designed using the Waterfall method including analysis, design, coding, testing, implementation. In the implementation process the system is built using MySQL database and PHP CodeIgniter framework as the programming language. Several unique features is also developed to the system such as location map to help *Customers* identify the store location and embed messaging application to facilitate *Customer* communication. To evaluate the system, we conduct several tests such as *black box* and SUS testing and system usability scale. Experimental results show that our system is well performed and meet the *user* expectation with high usability scale around 76%.

**Keywords:** Betta fish, web, Waterfall, PHP

## **1. PENDAHULUAN**

Ikan cupang merupakan ikan hias air tawar yang mudah untuk dipelihara dan dibudidayakan karena jenis ikan ini tidak memerlukan wadah yang besar dan makanan yang cukup mudah di dapat. Ikan hias mempunyai peran penting dalam menambah kesegaran, keindahan dan kesejukan lingkungan melalui jenis, warna, ukuran dan bentuk tubuhnya yang indah dan menarik (Mukti et al. 2019). Salah satu keistimewaan ikan cupang yaitu memiliki daya tahan tubuh yang kuat yang dapat hidup dengan wadah toples kecil dan kondisi air yang minim oksigen sekalipun, ikan ini tidak membutuhkan aerator atau gelembung udara seperti yang digunakan pada ikan hias lain, sehingga ikan ini banyak disukai masyarakat dari anak kecil hingga orang dewasa, selain itu ikan ini banyak digemari karena keindahan bentuk sirip dan warnanya, ikan cupang memiliki jenis yang banyak mulai *Half-moon*, *Plakat*, *Giant* dll, dan jenis tersebut masih mempunyai jenis lagi seperti *multicolor*, *blue rim*, *avatar* dan masih banyak lagi.

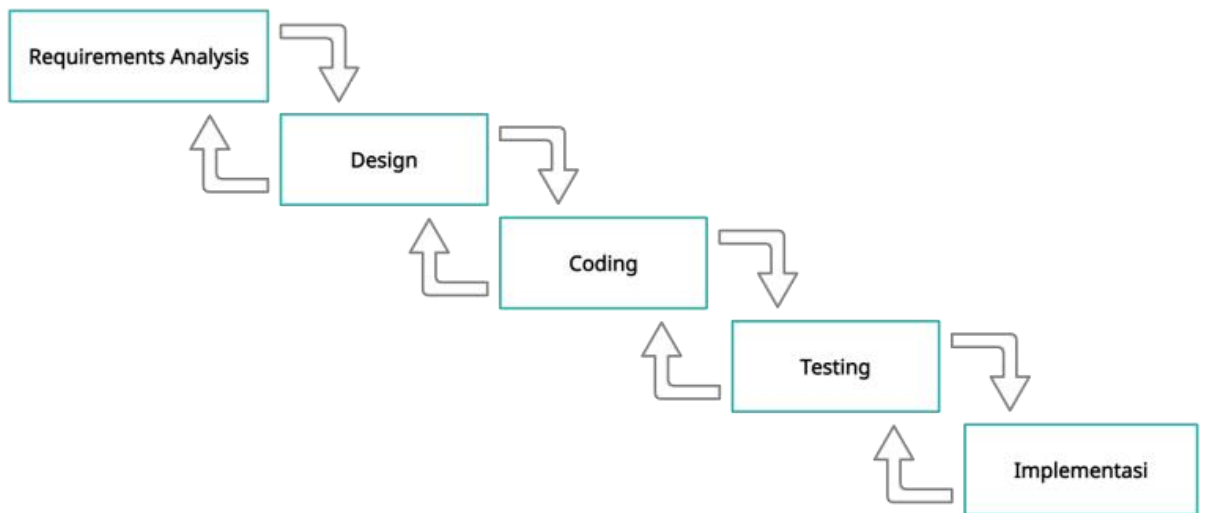
Ikan cupang menjadi tren saat ini, hal ini dilatar belakangi oleh banyaknya masyarakat yang menghabiskan waktunya dirumah, sehingga hobi merawat ikan cupang digandrungi oleh banyak orang, bagi masyarakat yang suka memelihara ikan cupang di daerah surakarta, toko ikan cupang labetaa solo yang berada di kecamatan Jebres, kota Surakarta, menjadi salah satu toko pilihan masyarakat daerah surakarta, toko labetaa solo ini menyediakan beragam jenis ikan cupang dan segala perlengkapan dan makanan ikan cupang, untuk orang yang menggemari ikan cupang bisa datang ke labetta solo untuk membeli atau hanya melihat ikan cupang, akan tetapi saat masa pandemi seperti ini masyarakat dianjurkan untuk tetap dirumah (Dwijayanti and Pramesti 2020). Mengabaikan tidak ada pembeli yang datang atau membeli ikan dengan penjualan offline, penjualan ikan dilabetaa solo harus menggunakan cara lain, seperti menjual melalui media social (Bahrudin, Sutomo, and Indonesia 2021). cara ini dinilai kurang efektif karena postingan di social media sering dihapus oleh *developer* karena dinilai kurang memenuhi kriteria di media sosial, karena saat penjual mengupload ikan cupang terutama ikan cupang jantan harus disampingkan dengan ikan cupang jantan lainnya agar sirip sirip ikan bisa berkembang untuk melihatkan keindahannya ke pembeli, hal itu dinilai tidak pantas untuk di tampilkan di social media.

Adanya masalah tersebut maka dibuatlah sistem informasi penjualan ikan cupang berbasis web, dengan sistem penjualan online yang dapat dikelola sendiri oleh pemiliknya, penjual tak perlu lagi khawatir untuk menjual ikan mereka. Adanya teknologi seperti ini dapat membantu dan mempermudah dalam penyelesaian pekerjaan (Sudarmilah, Supardi, and Muliawan 2012). penjualan online atau *e-commerce* sudah menjadi tren saat ini, penjualan online membuat para penjual tetap bisa mendapatkan penghasilan melalui platform *e-commerce* (Tran 2021), sebuah perusahaan pemasaran *e-bisnis*, memperkirakan penjualan *e-commerce* ritel di seluruh dunia akan meningkat dari 2,29\$ triliun pada tahun 2017 menjadi 4,48\$ triliun pada akhir tahun 2021 (Hwangbo, Kim, and Cha 2018), maka dari itu bisnis jual ikan ini bisa kita jadikan bisnis online agar bisnis ikan ini lebih meningkat dan dapat mencapai pasar global (Khoironi and Saskara 2015)(Destriana 2019). Diterapkannya system informasi penjualan ikan cupang berbasis web di labetta solo diharapkan dapat menarik minat pembeli dan memperluas jangkauan pasar mereka (Khomariah 2021), dengan kemudahan proses pembelian menggunakan system ini.

Sistem tersebut nantinya akan dikembangkan menggunakan metode waterfall meliputi *Requirements Analysis, Design, Implementation, Testing, Maintenance system*, system ini akan dibangun menggunakan tool seperti XAMPP, database MySQL dan PHP sebagai bahasa pemrograman, bahasa pemrograman PHP merupakan bahasa pemrograman yang tidak berbayar (*open source*) yang dapat digunakan oleh siapa saja. PHP juga dapat terintegrasi dengan baik pada database MySQL, aplikasi web berbasis PHP tergolong aman dibandingkan dengan aplikasi web yang dikembangkan menggunakan bahasa pemrograman lain (Putri, Rifnaldi, and Surmiyanti 2019).

## **2. METODE**

Metode penelitian yang diterapkan pada pembuatan tugas akhir ini adalah dengan pengembangan metode waterfall (Royce 1970). Metode waterfall merupakan model pengembangan sistem informasi yang sistematis dan sekuensial. Metode Waterfall memiliki tahapan-tahapan sebagai berikut:



Gambar 1. Metode Waterfall

## 2.1 *Requirements Analysis*

Proses *requirements analysis* yaitu mengumpulkan data-data mengenai sistem informasi penjualan ikan cupang, dalam pengumpulan data ini dapat dilakukan dengan cara wawancara kepada pemilik usaha untuk mengetahui proses atau cara yang digunakan oleh pemilik untuk menjual ikan cupang, observasi maupun penelitian secara langsung untuk mendapatkan data atau informasi yang lebih akurat, dalam tahap analisis nantinya akan didapatkan informasi mengenai penjualan ikan cupang, agar nantinya *software* yang dibuat dapat berjalan sesuai yang diharapkan. Proses wawancara didapatkan hasil bahwa terdapat 2 jenis *user* yaitu *Administrator* dan *Costumer*.

2.1.1 *Administrator* memiliki *User Accessibility* yaitu : sistem *login*, mengelola data transaksi, cek konfirmasi pembayaran, mengelola pengiriman barang, mengelola laporan transaksi, mengelola data produk, dan mengelola data *user*, mengelola kebutuhan sistem.

2.1.2 *Customer* memiliki *User Accessibility* yaitu : menampilkan dan menambahkan produk ke keranjang, sistem registrasi dan *login*, melakukan transaksi, mengelola account *user*, konfirmasi pembayaran, melakukan ulasan, menambahkan favorite.

## 2.2 *Design*

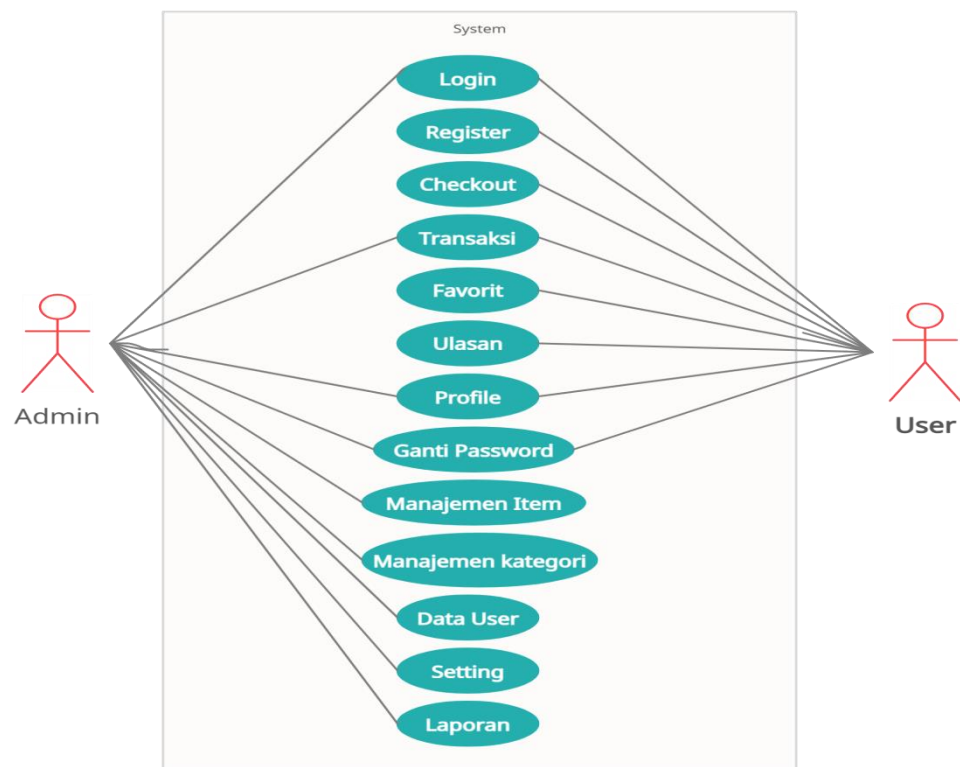
Tahapan ini berfokus pada pembentukan arsitektur keseluruhan dari sistem yang akan dibuat dapat berupa struktur data, perancangan *interface*, perancangan fungsi internal



maupun eksternal (Cahaya and Triayudi 2021). Tahap ini peneliti menggambarkan diagram yaitu *Use Case*, *Activity Diagram*, ERD (*Entity Relationship Diagram*) dan *User Interface*.

### 2.2.1 Use Case

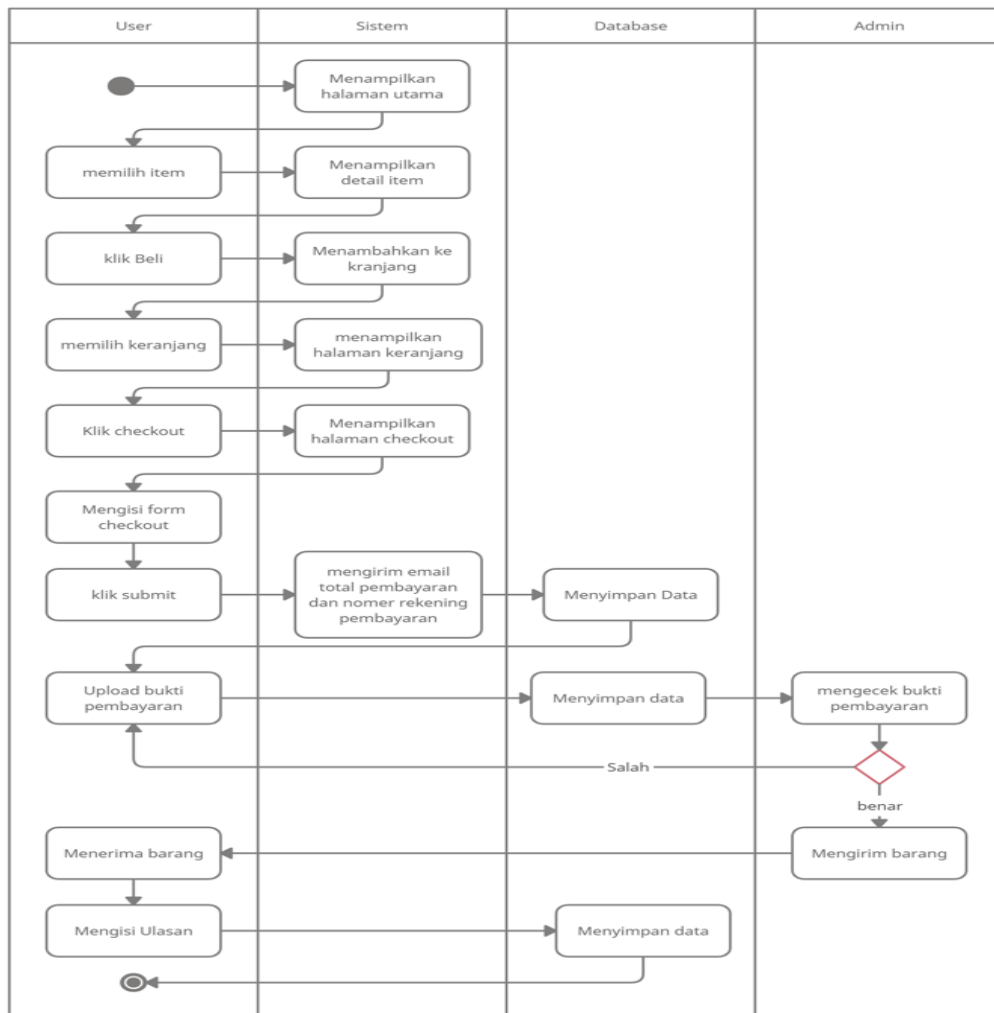
*Use case* merupakan gambaran dari *user* dan sistem yang mendeskripsikan sebuah interaksi dari satu *user* atau lebih dari sistem yang akan dibuat. Sebuah *use case* juga dapat diketahui interaksi apa saja yang akan dijalankan oleh sistem yang dibuat (Essebaa and Chantit 2018).



Gambar 2. Use Cas

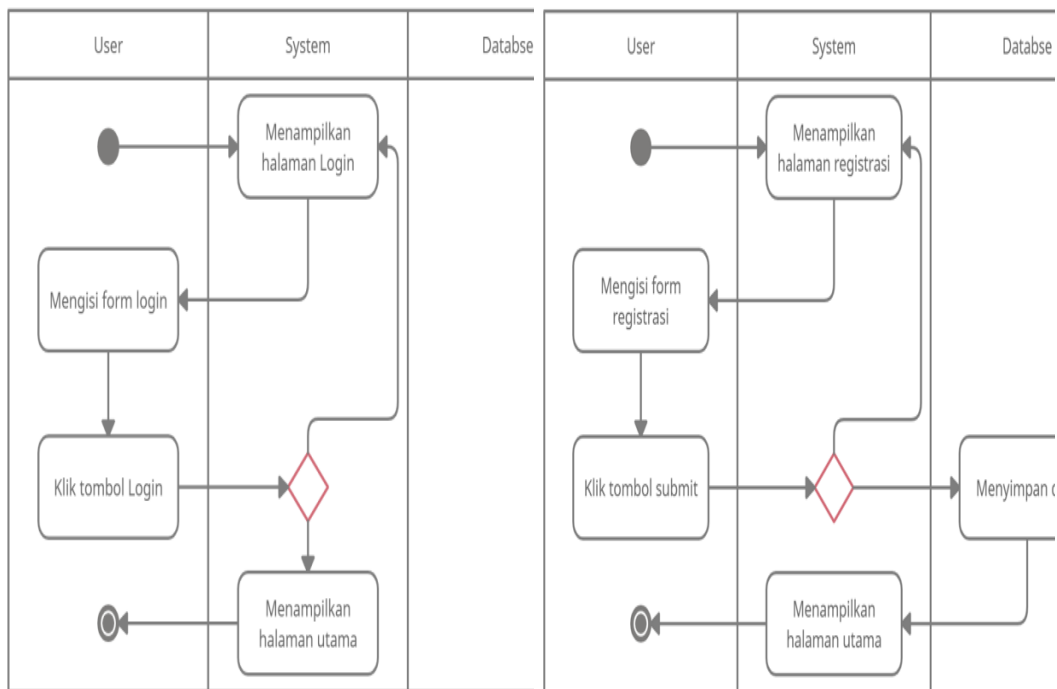
### 2.2.2 Activity Diagram

*Activity Diagram* dibuat untuk menjelaskan proses yang dilakukan antara interaksi *user* pada sistem dimulai dari awal sampai akhir proses, *Activity Diagram* juga menggambarkan jalannya proses sebuah sistem secara garis besar. Gambar 3 merupakan *Activity Diagram Admin*. Gambar 4 merupakan *Activity Diagram User* atau pengguna.



Gambar 3. Activity Diagram Transaksi

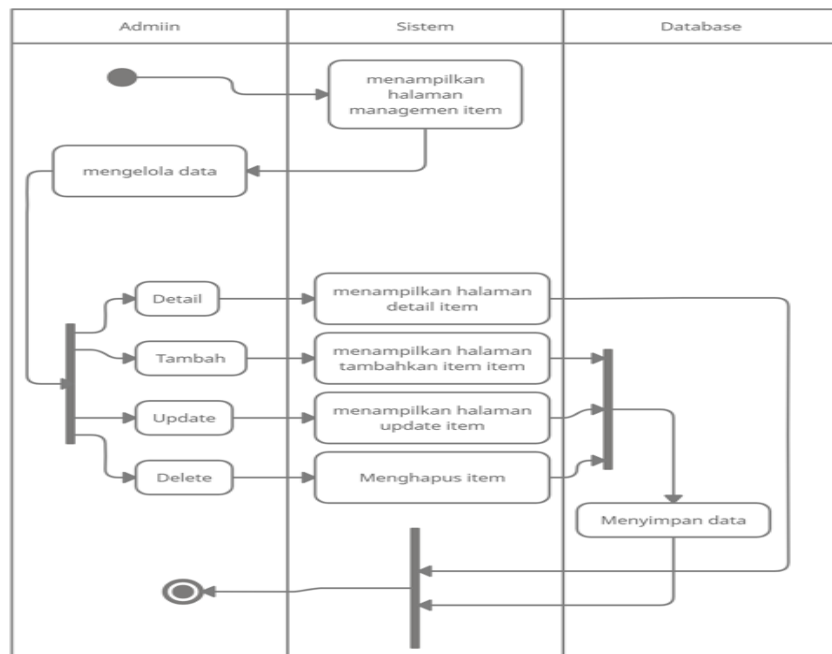
Gambar 3. Ketika *user* berhasil menambahkan item ke keranjang lalu mengklik *checkout* maka akan masuk ke halaman *checkout*, muncul total pembayaran setelah *user* mengisi alamat pengiriman lalu klik *submit*, *User* akan menerima email *invoice* pembayaran dan nomer rekening pembayaran, kemudian *user* mengupload bukti pembayaran di halaman upload bukti, kemudian admin akan mengkonfirmasi pembayaran dan memproses pengiriman barang jika pembayaran valid, *Administrator* akan mengirimkan bukti pengiriman yang dapat dilihat di detail transaksi *user*, *user* mengkonfirmasi barang telah diterima jika barang sudah sampai kemudian mengisi ulasan barang.



Gambar 4. Activity Diagram Login      Gambar 5. Activity Diagram Register

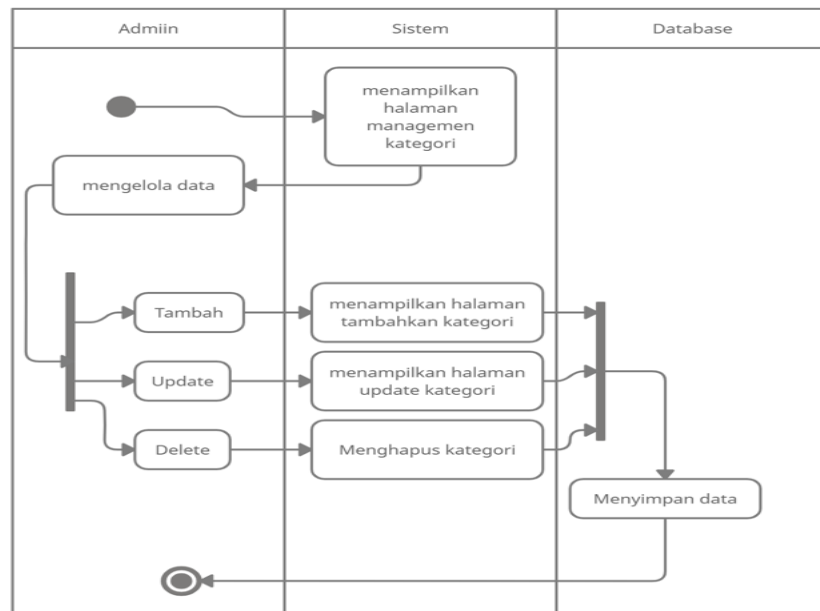
Gambar 4. Merespon permintaan *user* dengan menampilkan halaman *Login* kemudian *user* menginputkan email dan *password* lalu klik tombol '*Login*', maka sistem akan melakukan *otentifikasi* dengan database, jika berhasil akan diarahkan langsung ke halaman utama atau dashboard *administrator*, jika gagal akan tetap berada di halaman *login*.

Gambar 5. Merespon permintaan *user* dengan menampilkan halaman registrasi kemudian *user* menginputkan form registrasi kemudian klik '*Submit*', maka sistem akan melakukan otentifikasi dengan database, ketika berhasil akan diarahkan langsung ke halaman *login*, jika gagal akan tetap berada di halaman registrasi serta muncul pesan kesalahan pada inputan *user*.



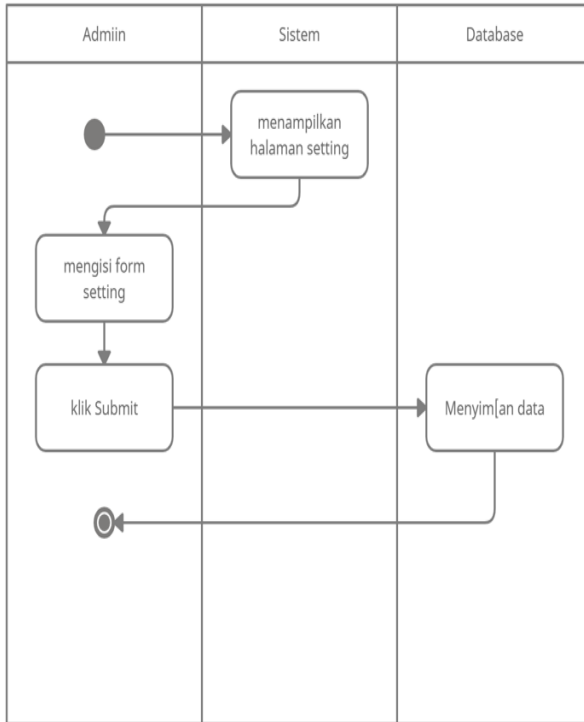
Gambar 6. Activity Diagram Manajemen item

Gambar 6. Dalam mengelola data Item, *administrator* akan mengakses halaman Manajemen Item lalu sistem akan merespon dengan menampilkan data item. *Administrator* mempunyai Hak untuk melakukan manajemen data item.

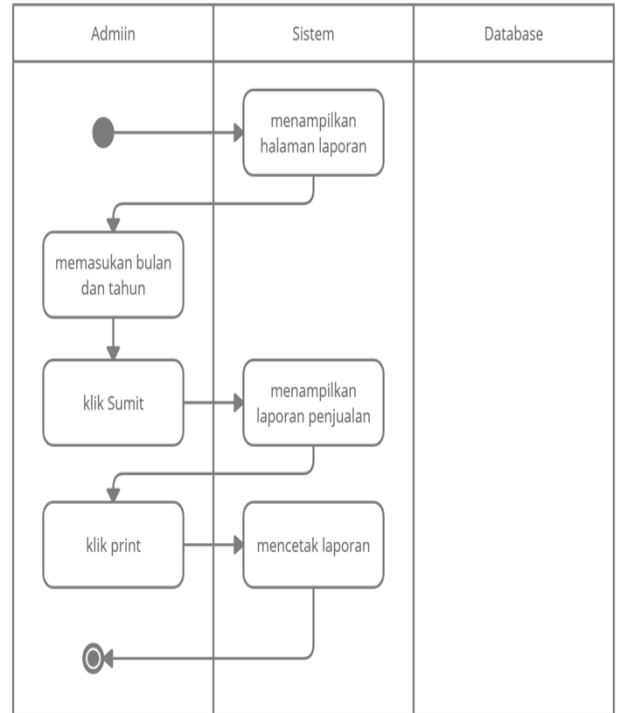


Gambar 7. Activity Diagram Manajemen Kategori

Gambar 7. Dalam mengelola data Kategori, *Administrator* akan mengakses halaman Management Kategori lalu sistem akan merespon dengan menampilkan data Item. *Administrator* mempunyai Hak untuk melakukan manajemen data Kategori

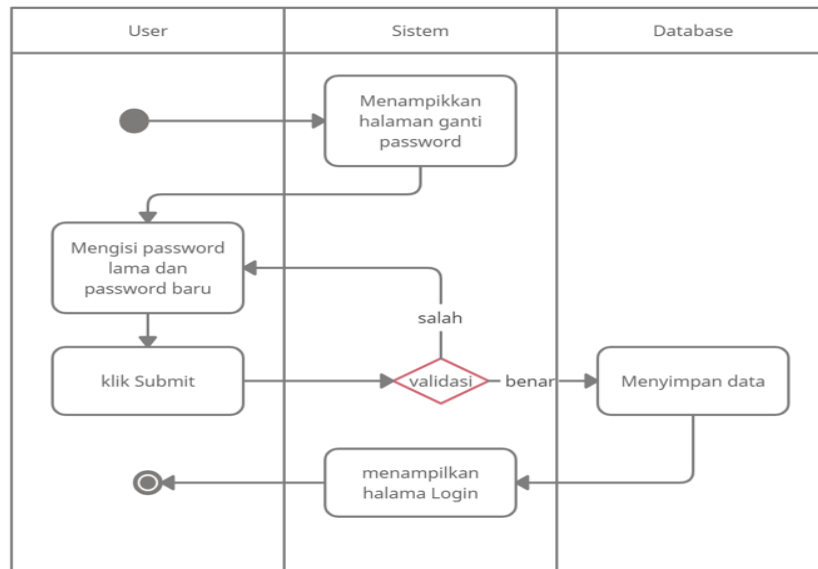


Gambar 8. Activity Diagram Setting



Gambar 9. Activity Diagram Laporan

Gambar 8. Mengelola data web, *administrator* akan mengakses halaman Setting lalu sistem menampilkan data Setting. *administrator* mempunyai Hak untuk mengubah data website seperti email, no tlp dan seainya. Gambar 9. mengelola Laporan penjualan, *administrator* akan mengakses halaman Laporan. *Administrator* dapat mengetahui laporan penjualan setiap bulan dan mencetak atau print data penjualan.

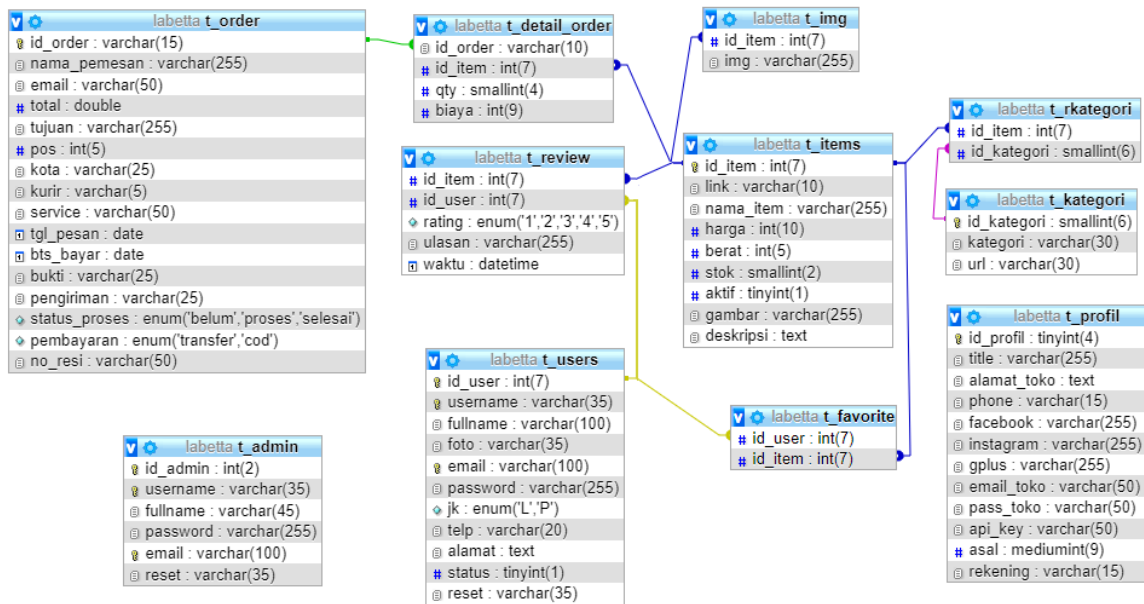


Gambar 10. Activity Diagram Ganti Password

Gambar 10. Activity ganti password, administrator atau user akan mengakses halaman Ganti Password kemudian sistem akan merespon dengan menampilkan form ganti password. Dimana administrator dan user mempunyai hak untuk mengganti password akun miliknya.

### 2.2.3 Entity Relationship Diagram (ERD)

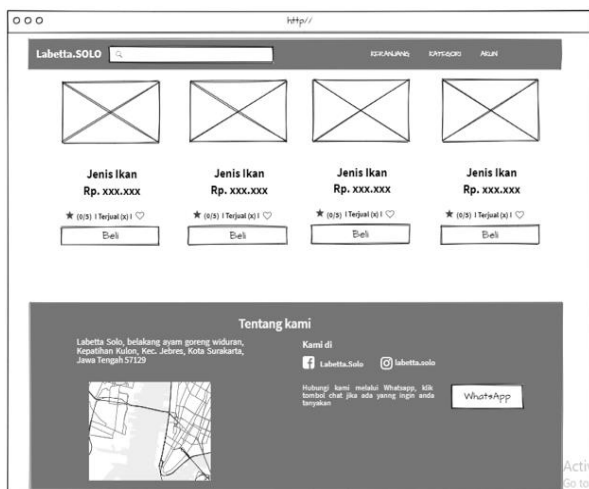
Entity Relationship Diagram (ERD) adalah penggambaran suatu rancangan basis data yang melibatkan hubungan atau relasi antar entitas atau object yang terlihat beserta atributnya, gambar 11 merupakan gambaran dari entity relationship diagram penjualan ikan cupang di labetta solo.



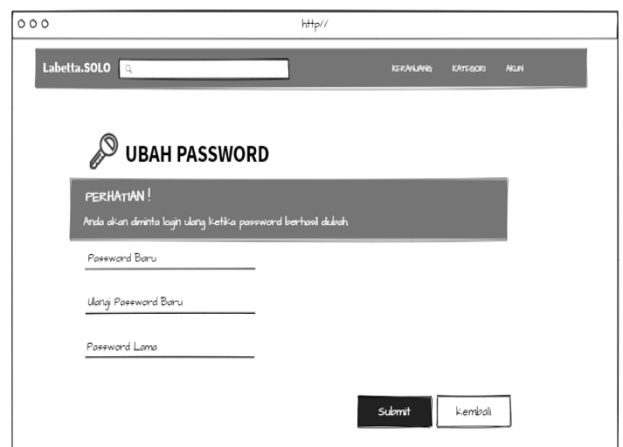
Gambar 11. Entity Relationship Diagram

### 2.2.4 Design User Interface

User Interface adalah tampilan visual sebuah produk yang menjembatani sistem dengan pengguna (*user*). Tampilan UI dapat berupa bentuk, warna, dan tulisan yang didesain semenarik mungkin. Secara sederhana, UI adalah bagaimana tampilan sebuah produk dilihat oleh pengguna. Proses pembuatan *design user interface* ini menggunakan tool MockFlow. Gambar dibawah merupakan *desain User Interface* dari sistem penjualan ikan cupang di Labetta Solo.



Gambar 12. Halaman Utama User



Gambar 13. Halaman Ubah Password

Gambar 12 adalah tampilan utama pada saat *user* masuk kedalam sisitem, tampilan ini berisikan daftar produk, maps atau alamat toko, serta terdapat tombol

WhatsApp yang akan terhubung dengan whatsapp admin. Gambar 13 adalah tampilan saat *user* masuk ke dalam halaman ubah *password*, *user* harus menginputkan *password* baru dan *password* lama untuk merubah *password*.

Gambar 14. Halaman Registrasi *User*

Gambar 15. Halaman *Login*

Gambar 14 merupakan halaman registrasi atau pendaftaran bagi *user*, *user* harus mengisikan form pendaftaran agar *user* terdaftar dan mendapatkan akses masuk ke sistem. Gambar 15 adalah halaman *login* untuk masuk ke sistem dengan mengisikan *username* dan *password* yang terdaftar jika data yang dimasukkan sesuai maka sistem akan menampilkan halaman utama.

Gambar 16. Halaman *Checkout User*

Id Transaksi	Tgl Pesan	Tgl Bayar	total Bayar	Status	Opsi
00000000	xxxxxxxx	xxxxxxxx	Rp xxxxxxxx	Diproses	View
1. 00000000	xxxxxxxx	xxxxxxxx	Rp xxxxxxxx	Di Kirim	View

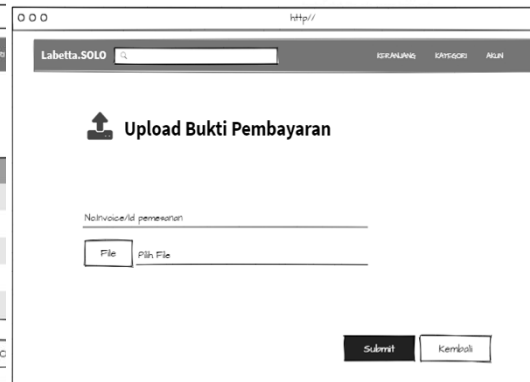
Gambar 17 Halaman *Transaksi User*



Gambar 16 merupakan halaman *checkout user* untuk mengisi data alamat pengiriman, kurir dan total belanja yang harus dibayarkan oleh pembeli. Gambar 17 halaman Transaksi *user* laman ini berupa daftar informasi transaksi yang dilakukan oleh *user*.

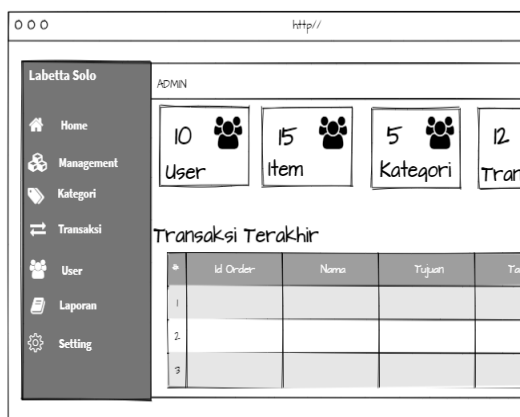


Gambar 18. Halaman Keranjang

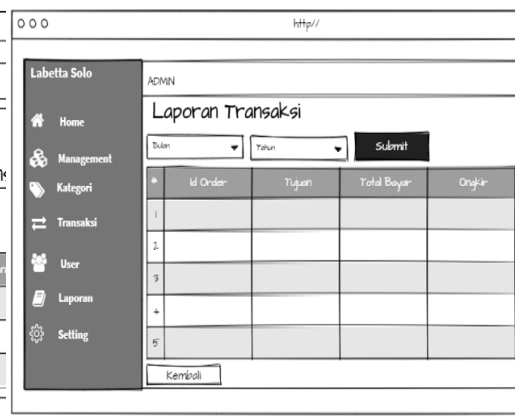


Gambar 19. Halaman Upload Bukti Bayar

Gambar 18 Halaman keranjang *user* merupakan halaman daftar transaksi yang dilakukan oleh *user*. Gambar 19 halaman upload bukti pembayaran merupakan halaman untuk mengupload struk bukti transfer dengan memasukkan *No Invoice* atau id pesanan dan upload foto struk bukti transfer.



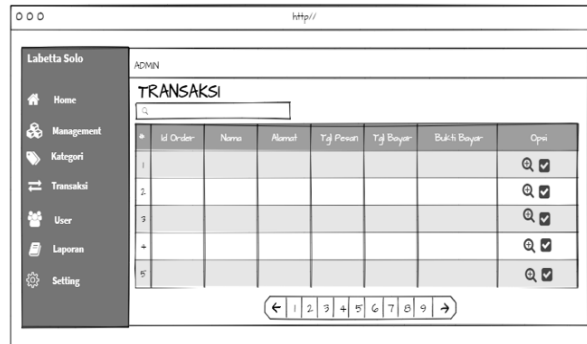
Gambar 20. Home Admin



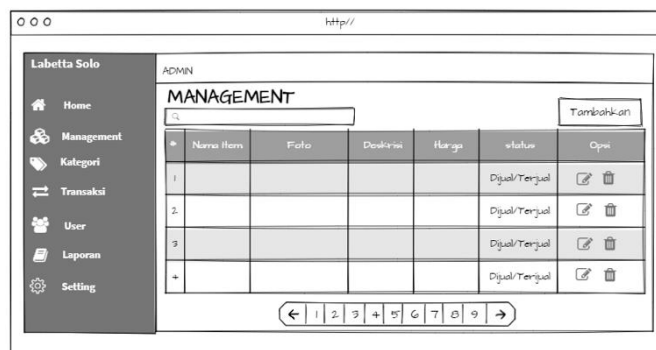
Gambar 21. Halaman Laporan

Gambar 20. tampilan *home* admin merupakan tampilan saat admin berhasil *login*, tampilan ini berisikan riwayat transaksi terakhir, jumlah *user*, item kategori dan

transaksi berhasil. Gambar 21 halaman laporan merupakan halaman hasil laporan penghasilan dari transaksi penjualan ikan, laporan dapat dicetak bulanan.

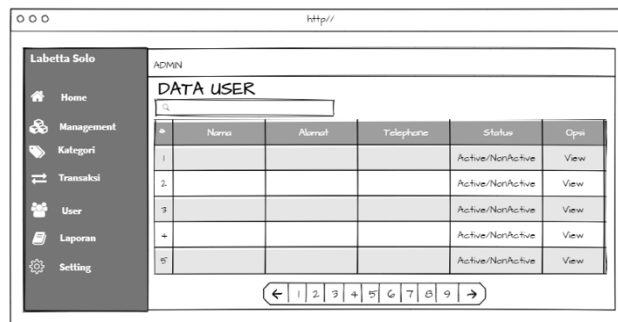


Gambar 22. Halaman Transaksi Admin

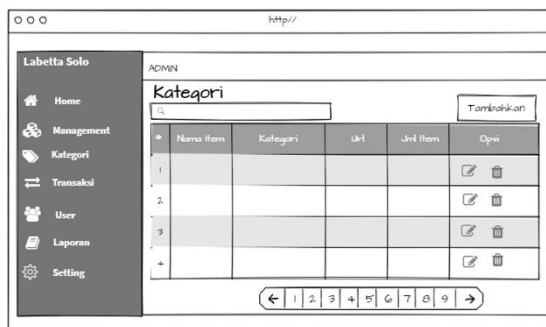


Gambar 23. Halaman Pengelolaan Data

Gambar 22 halaman transaksi admin berisikan informasi data pesanan dari *user*, admin dapat melihat detail transaksi dan mengkonfirmasi proses transaksi. Gambar 23 halaman pengelolaan data, pada halaman ini admin dapat mengelola data ikan yang dijual seperti menambahkan, edit dan hapus data.



Gambar 24. Halaman Kategori



Gambar 25. Halaman Data *User*

Gambar 24 halaman kategori merupakan halaman management data kategori admin, admin memiliki akses CRUD pada halaman kategori. Gambar 25 halaman data *user* yang berisikan seluruh informasi data mengenai *user* yang telah terdaftar.

### 2.3 Coding

Tahap ini merupakan tahap pembuatan aplikasi dengan menggunakan php sebagai bahas pemrograman dan penggunaan tool Framework CodeIgniter 3(PHP), Sublime Text Editor, XAMPP, *database* MySQL,

Arsitektur yang digunakan pada Codeigniter dibagi menjadi 3 komponen yaitu MVC (*Model, View, Controller*). Sehingga setiap komponen memiliki file codingan masing-masing yang saling terhubung.

#### 2.3.1 Model

Bertugas untuk mengatur, menyiapkan, memanipulasi dan mengorganisasikan data (dari database) sesuai dengan instruksi dari *controller*. Sehingga terdapat codingan :

- a. Admin.php menerima respons dari *controller* yang terkait dengan tabel Admin atau lebih dari satu tabel tetapi memiliki fungsi yang sama dalam mengelola data admin.
- b. Items.php menerima respons dari *controller* yang terkait dengan tabel Items atau lebih dari satu tabel tetapi memiliki fungsi yang sama dalam mengelola data item.
- c. Trans.php menerima respons dari *controller* yang terkait dengan tabel order atau lebih dari satu tabel tetapi memiliki fungsi yang sama dalam mengelola data transaksi.
- d. Users.php menerima respon dari *controller* yang berhubungan dengan tabel *User* ataupun lebih dari satu tabel tetapi memiliki function yang sama dalam mengelola data *User*.

### 2.3.2 View

Bertugas untuk menyajikan informasi (yang mudah dimengerti) kepada *user* sesuai dengan instruksi dari controller. Terdapat 2 *Views* yaitu untuk *Administrator* dan *Customer*.

#### a. *Administrator*

- 1) *Home.php* menampilkan halaman *Home* admin.
- 2) *Laporan.php* menampilkan laporan penjualan.
- 3) *Manage\_item.php* menampilkan halaman untuk mengelola data item.
- 4) *Tag.php* menampilkan halaman untuk mengelola data kategori.
- 5) *Transaksi.php* menampilkan halaman transaksi admin.
- 6) *Manage\_user.php* menampilkan halaman data *user*.

#### b. *Customer*

- 1) *Pass.php* menampilkan halaman ubah *password*.
- 2) *Checkout.php* menampilkan form pengisian alamat dan metode pembayaran.
- 3) *Login.php* menampilkan halaman *login*.
- 4) *Register.php* menampilkan halaman registrasi.
- 5) *Up\_bukti.php* menampilkan halaman untuk mengupload bukti sudah dilakukannya pembayaran.
- 6) *Review\_data.php* menampilkan halaman *review* data barang.
- 7) *User\_profil.php* menampilkan halaman data *user*.

2.3.3 Controller, Bertugas untuk mengatur apa yang harus dilakukan model, dan *view* mana yang harus ditampilkan berdasarkan permintaan dari *user*. Namun, terkadang permintaan dari *user* tidak selalu memerlukan aksi dari model.

- a. *Checkout.php* merespon request dari *user* yang berhubungan dengan *checkout*.
- b. *Item.php* merespon request dari *user* dalam mengelola data item entah untuk menampilkan detail data item atau CRUD terhadap data item.
- c. *Login.php* merespon request dari *user* yang berhubungan dengan *Login* dan *Logout* sistem.
- d. *Setting.php* merespon request dari *user* yang berhubungan dengan *setting* dalam mengelola website admin entah untuk menampilkan data atau CRUD terhadap data pada website admin.
- e. *Transaksi.php* merespon request dari *user* yang berhubungan dengan *user* dalam

mengelola *user* entah untuk menampilkan detail data *user* atau CRUD terhadap data *user*.

- f. *User.php* merespon request dari *user* yang berhubungan dengan *user* dalam mengelola *user* entah untuk menampilkan detail data *user* atau CRUD terhadap data *user*.

Selain itu terdapat *software third-parties* yang digunakan untuk mengembangkan program serta mendukung proses yang dilakukan oleh *user*. Sistem dikembangkan mengacu pada kemampuan *user Administrator* dan *Customer* dalam mengaplikasikannya. Langkah ini diambil guna mempermudah *Administrator* dan Pembeli dalam penggunaan sistem pada saat di implementasikan.

## **2.4 Testing**

Hasil code yang sudah jadi selanjutnya memasuki tahap testing atau pengujian yang bertujuan untuk sejauh mana sistem yang dibuat dapat berjalan, tahap testing ini akan menggunakan metode *kuesioner* dan *black box*, Pengujian *kuesioner* dilakukan dengan SUS (*System Usability Scale*). Penelitian ini menggunakan metode *System Usability Scale* untuk mendapatkan perspektif pengguna agar hasil akhir yang didapat sesuai dengan kebutuhan, dengan kata lain apabila pengguna menyatakan fungsi telah sesuai dengan apa yang dipahami oleh pengguna, maka fitur aplikasi telah berfungsi dengan baik. *black box* digunakan untuk memastikan spesifikasi dan sistem tidak mengalami kesalahan atau eror pada output dari input yang diberikan. Pengujian sistem akan dilakukan menggunakan perangkat PC atau Laptop yang sudah terinstall XAMPP sebagai local server. Dalam tahap ini belum dilakukan hosting karena masih dalam tahap pengujian. Hasil pengujian dijadikan sebagai acuan dalam perbaikan sebelum sistem diimplementasikan.

## **2.5 Implementasi**

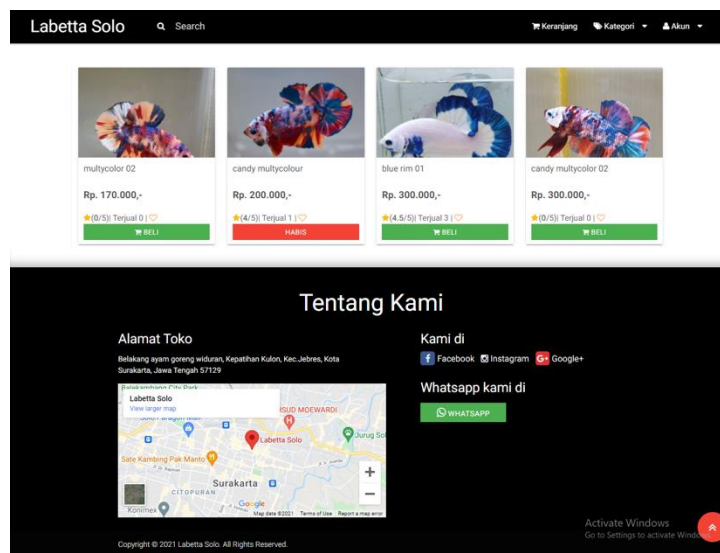
Tahap terakhir yaitu sistem yang telah dibuat dan telah di uji maka sistem akan digunakan oleh pemilik labetta solo, sistem yang sudah sempurna dapat di hosting dengan menyewa jasa *hosting* berbayar seperti Hostinger Indonesia, Niagahoster, SiteGround agar keamanan sistem dapat terjaga lebih baik, selain itu sistem ini akan mendapatkan pemeliharaan seperti memperbaiki kesalahan sistem yang ditemukan oleh *user*. sistem akan dijalankan pada perangkat keras PC (*Personal Computer*) atau

menggunakan laptop melalui browser diantaranya Google Chrome, Opera, Mozilla Firefox atau browser lain yang tersedia.

### 3. HASIL DAN PEMBAHASAN

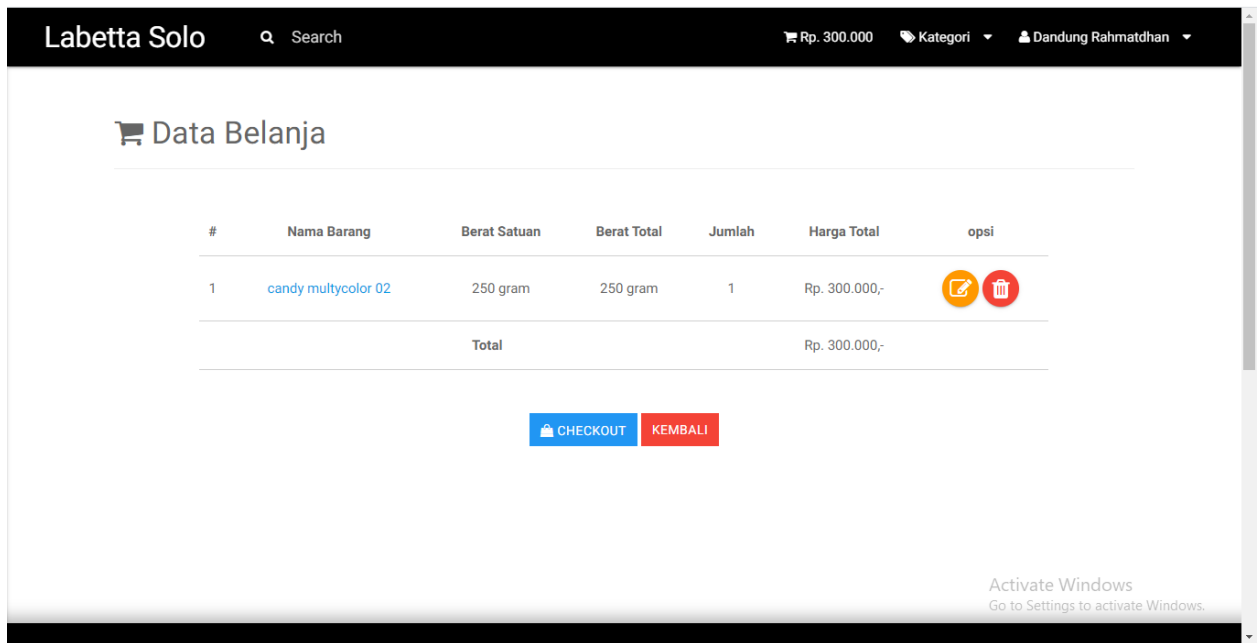
Penelitian ini menghasilkan sistem informasi penjualan ikan cupang berbasis web pada Labetta Solo sistem yang dihasilkan memiliki *interface* sesuai yang diharapkan, sistem ini juga dapat melaporkan hasil penjualan secara otomatis dan juga penjualan ikan di labetta solo.

#### 3.1 Hasil Penelitian



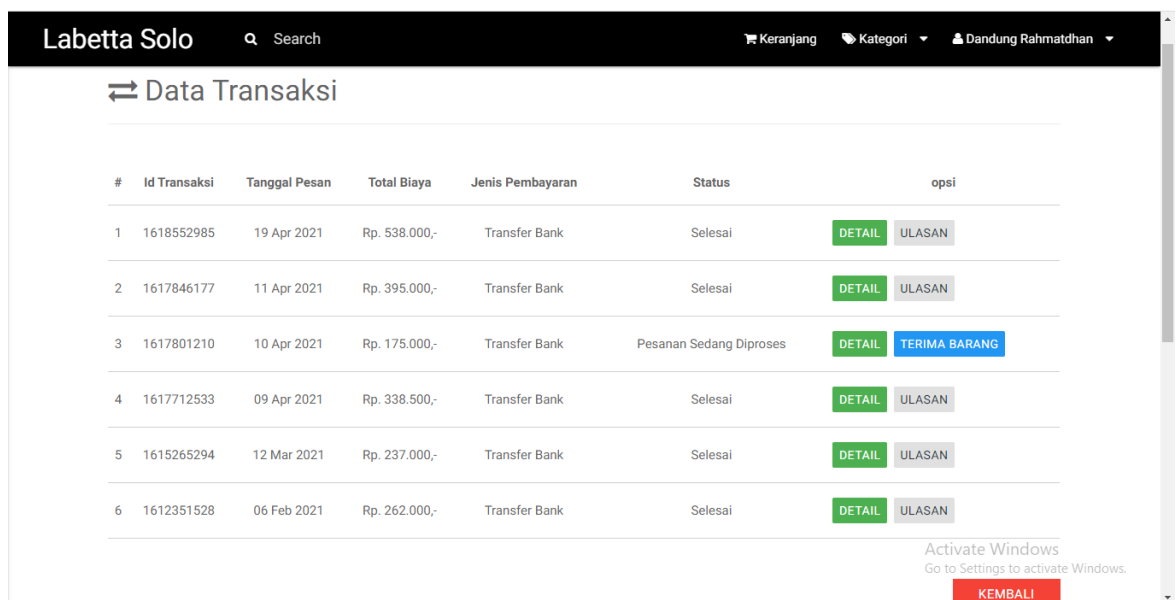
Gambar 26. Home User

Gambar 26 Tampilan halaman utama dari *user*, disini *user* dapat memilih ikan dan membeli ikan dengan mengklik tombol beli jika stok masih ada maka istem akan ditambahkan ke keranjang, untuk melihat detail item *user* bisa mengklik gambar maka akan masuk ke halaman detail item. Terdapat rating penjualan, jumlah terjual dan love untuk menambahkan favorit ke dalam daftar favorite *user*.



Gambar 27. Keranjang

Gambar 27 merupakan halaman keranjang yang menampilkan daftar data belanja *user* di halaman ini *user* dapat menghapus atau mengubah data belanja.



Gambar 28. Transaksi *User*

Gambar 28 Halaman Transaksi *user* berisikan daftar transaksi yang dilakukan oleh *user*, terdapat status untuk melihat proses dari transaksi pembelian dan beberapa opsi seperti detail untuk melihat detail pembelian dan bukti pengiriman, Terima barang

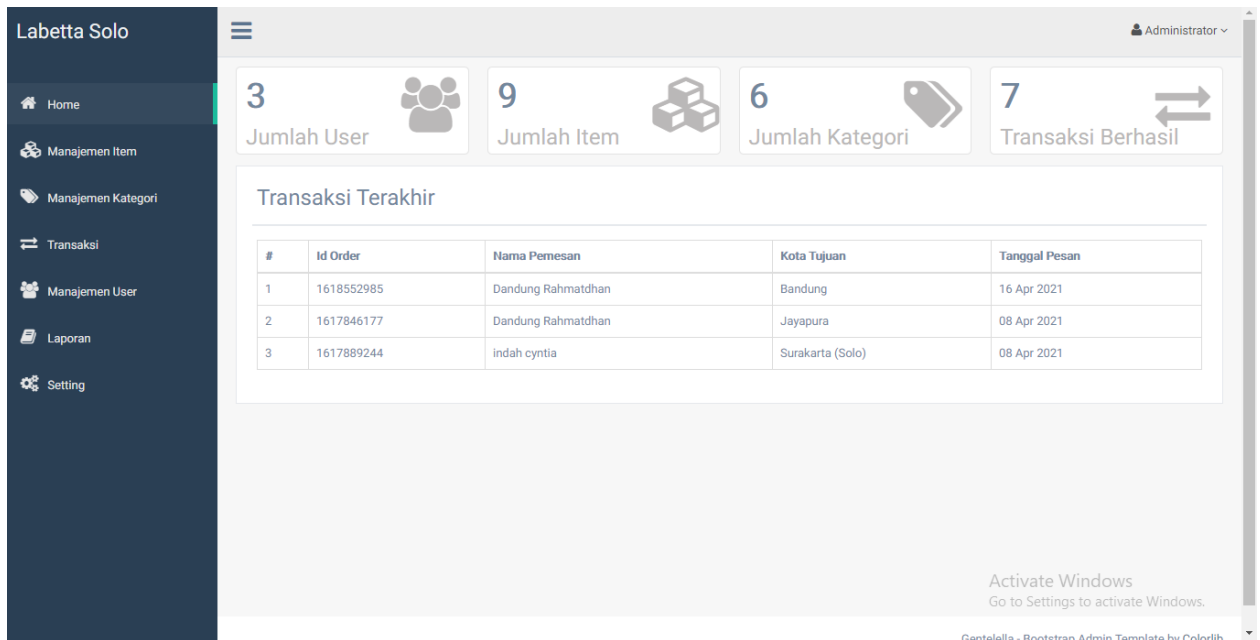
untuk mengkonfirmasi ke admin bahwa barang sudah diterima, dan Ulasan untuk berikan penilaian produk yang dibeli.

The screenshot shows the checkout interface for 'Labetta Solo'. The header features the store name, a search bar, a shopping cart icon with a value of 'Rp. 300.000', and user information 'Dandung Rahmatdhan'. The main content area is titled 'Checkout' and contains several form elements: a 'Provinsi' dropdown menu, a 'Pilih Kota / Kabupaten' dropdown menu, an 'Alamat' input field, a 'Kode Pos' input field, a 'Pilih Kurir' dropdown menu with 'POS' selected, a 'Pilih Layanan' dropdown menu, an 'Ongkos Kirim' input field with '0' entered, a 'Pilih Pembayaran' dropdown menu with 'Pilih Metode Pembayaran' selected, and a 'Total Biaya' input field with '300.000' displayed. A red asterisk note indicates '\* (COD) hanya untuk wilayah SURAKARTA/SOLO'. At the bottom of the form are two buttons: 'SUBMIT' and 'KEMBALI'. A Windows watermark is visible on the right side of the page.

Gambar 29. *Checkout User*

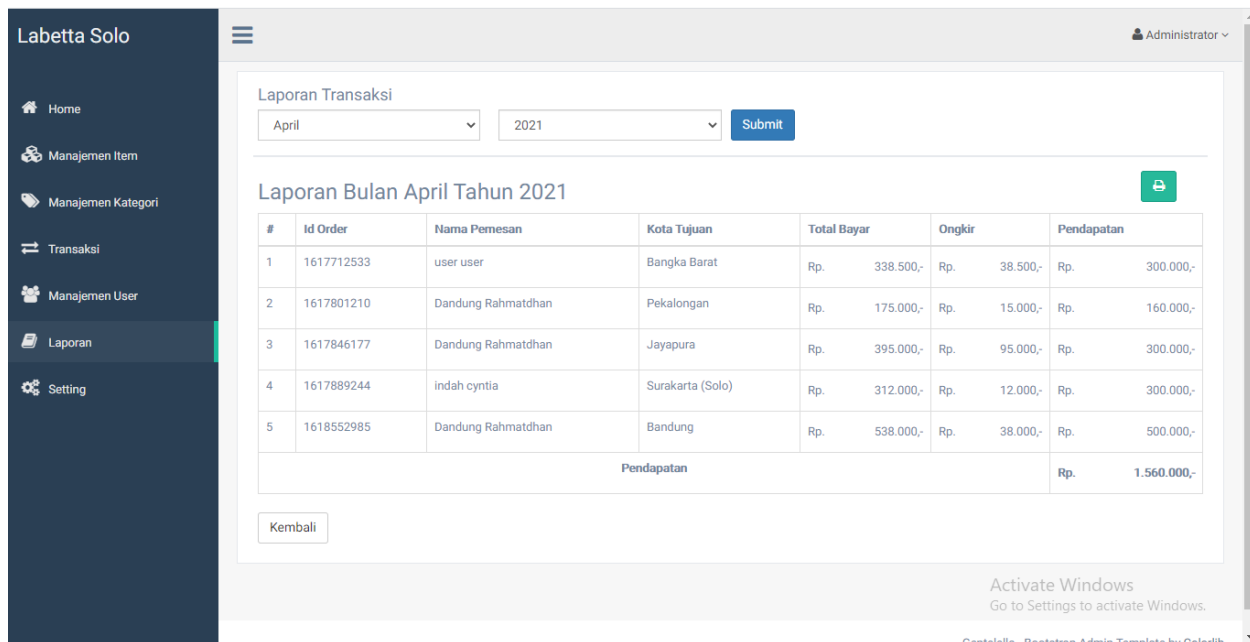
Gambar 29. Halaman *checkout user* berisikan data alamat pengiriman dan opsi kurir pengiriman untuk menghitung ongkos kirim, secara otomatis ongkos kirim akan ditambahkan dengan harga barang yang dibeli, untuk metode pembayaran *user* bisa memilih COD (*Cash On Delivery*) untuk *Customer* yang berada di wilayah Surakarta dan Transfer untuk luar kota Surakarta.





Gambar 30. Halaman *Home* Admin

Gambar 30. Halaman *Home* admin berisikan daftar transaksi terakhir, jumlah *user*, jumlah item, jumlah kategori, dan jumlah transaksi berhasil



Gambar 31. Laporan Admin

Gambar 31. Halaman laporan admin melaporkan hasil penjualan selama periode bulan maka akan muncul pendapatan jumlah penjuala ikan selama periode satu bulan, admin dapat mencetak laporan untuk bukti fisik hasil laporan penjualan.

### 3.2 Pengujian *Black box*

Pengujian adalah suatu rangkaian aktifitas yang terencana dan sistematis untuk menguji atau mengevaluasi kebenaran yang diinginkan (Yulistina et al. 2020). Pengujian *Black box* merupakan metode untuk menguji sistem yang telah dibuat, untuk menentukan *bug* atau gangguan pada sistem sebelum sistem di implementasikan, tanpa harus memperhatikan detail *software* (Priyaungga et al. 2020). Adanya *software* testing dapat mengetahui apakah sistem tersebut sudah memenuhi semua kriteria yang dibutuhkan oleh *user*. Pengujian *black box* berfokus pada input yang tersedia dan output yang diharapkan tanpa pengetahuan tentang internal programnya. Hasil pengujian *black box* pada sistem penjualan ikan cupang berbasis web di Labetta Solo dapat dilihat pada Tabel 1 berikut.

Tabel 1. Pengujian Black Box

No	Pengujian	Kondisi Pengujian	Harapan	Hasil
1	<i>Login</i>	<i>Username</i> dan <i>Password</i> salah <i>Username</i> dan <i>password</i> benar	Kembali ke halaman <i>login</i> sistem masuk ke halaman utama	Berhasil
2	<i>Logout</i>	Keluar dari sistem dan menampilkan halaman <i>login</i>	Sistem berhasil keluar dan menampilkan halaman <i>login</i>	Berhasil
3	Registrasi	Mengakses <i>form registrasi</i> , menginputkan nama depan, nama belakang, email, dan <i>password</i> .	Email verifikasi terkirim ke email <i>user</i> dan mengarah ke halaman <i>login</i> .	Berhasil
4	Ganti <i>password</i>	Akses halam ganti <i>password</i> , masukan <i>password</i> baru, masukan <i>password</i> baru sekali lagi, masukan <i>password</i> lama, klik <i>submit</i>	Kembali kehalaman <i>login</i> untuk <i>login</i> menggunakan <i>password</i> baru	Berhasil
5	Menambahkan produk ke <i>Keranjang</i>	Mengakses halaman detail produk atau <i>quick view</i> , klik tombol <i>Beli</i>	Produk ditambahkan ke keranjang	Berhasil

6	Menghapus produk dari Keranjang	Klik halaman <i>keranjang</i> , lalu klik icon ' <i>Trash</i> '.	Produk hilang dari keranjang.	Berhasil
7	Pemesanan produk	Akses halaman Keranjang, lalu klik <i>Checkout</i> akan mengarah ke halaman <i>Checkout</i> , mengisi form yang berkaitan dengan informasi data diri dan pengiriman. Klik <i>Submit</i> .	Muncul notifikasi pemesanan berhasil dilakukan dan email pemberitahuan informasi mengenai order, terkirim ke email <i>user</i> .	Berhasil
8	Upload Bukti pembayaran	Mengakses halaman Bukti Pembayaran, lalu mengisi form <i>no invoice</i> atau id pesanan dan upload file gambar bukti transfer lalu klik tombol <i>submit</i> .	Muncul <i>alert</i> bahwa pengiriman berhasil	Berhasil
9	Ulasan	Mengakses halaman ulasan pada transaksi yang berhasil lalu mengisi atau edit bintang dan form keterangan.	Sistem berhasil menambahkan atau mengedit ulasan	Berhasil
10	Edit Data Pesanan	Mengakses halaman keranjang kemudian pilih icon edit, ubah jumlah pesanan lalu klik simpan	Muncul alert bahwa pesanan berhasil di update	Berhasil
11	Menambahkan Favorit	Klik icon love pada halaman utama atau pada detail produk.	Item berhasil ditambahkan ke favorit	Berhasil
12	Menghapus Favorite	Klik icon love pada halaman utama atau pada detail produk.	Item berhasil dihapus dari favorit	Berhasil
13	Favorite	Sistem menampilkan <i>view</i> data favorite	Sistem dapat menampilkan <i>view</i> data favorite	Berhasil
14	Halaman profil <i>user</i>	Menampilkan <i>view</i> data <i>user</i> dan	Sistem dapat menampilkan <i>view</i>	Berhasil

		melakukan edit data <i>user</i> .	data <i>user</i> dan melakukan edit data <i>user</i>	
15	Mengelola item	Menampilkan <i>view</i> data item dan melakukan CRUD pada data item	Sistem dapat menampilkan <i>view</i> data item dan melakukan CRUD pada data item	Berhasil
16	Mengelola kategori	Menampilkan <i>view</i> data kategori dan melakukan CRUD pada data kategori	Sistem dapat menampilkan <i>view</i> data kategori dan melakukan CRUD pada data kategori	Berhasil
17	Setting	Menampilkan <i>view</i> data setting dan melakukan edit pada data setting	Sistem dapat menampilkan <i>view</i> data setting dan melakukan edit pada data setting	Berhasil
18	Manajemen <i>user</i>	Menampilkan <i>view</i> data <i>user</i> dan mengubah status <i>user</i>	Sistem dapat menampilkan <i>view</i> data <i>user</i> dan mengubah status <i>user</i> .	Berhasil
19	Halaman Laporan	Menampilkan <i>view</i> data laporan sesuai periode yang diinginkan dan mencetak laporan penjualan	Sistem dapat menampilkan <i>view</i> data laporan sesuai periode yang diinginkan dan mencetak laporan penjualan	Berhasil
20	Konfirmasi pembayaran	Menampilkan halaman transaksi <i>view</i> data transaksi jika benar klik icon “centang”	Sistem mengubah ststus pembayaran pada <i>user</i>	Berhasil
21	Upload Bukti Pengiriman	Mengakses halaman bukti pengiriman, lalu mengisi form no resi dan upload file gambar bukti pengiriman lalu klik tombol <i>submit</i> .	Jika berhasil maka akan kembali ke halama transaksi	Berhasil
22	Chat	Klik tombol whatsapp	Masuk kehalaman wahtsapp web	Berhasil

Berdasarkan hasil pengujian *black box* pada tabel 1, dapat dinyatakan bahwa dari 22 poin yang diujikan dapat berjalan baik dengan fungsi dan fiturnya, sesuai dengan harapan pada rancangan awal sistem ini dibentuk. Sistem labetta solo tidak menutup kemungkinan jika terjadi kesalahan dan memerlukan perbaikan.

### 3.3 Pengujian System Usability Scale (SUS)

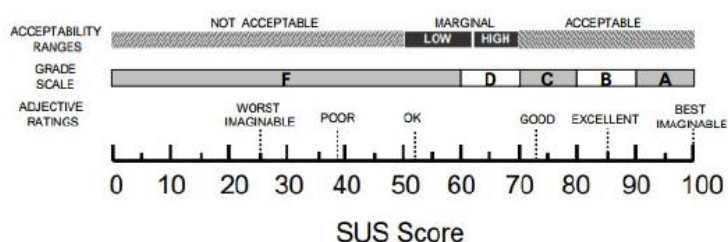
*Usability* testing umumnya diringkas sebagai kualitas kenyamanan sistem. Kegunaan meliputi efektifitas, efisiensi, dan kepuasan pengguna terhadap sistem (Sudarmilah and Siregar 2019). SUS (*System Usability Scale*) merupakan *kuesioner* yang dapat digunakan untuk mengukur usability sistem komputer menurut sudut pandang subyektif pengguna, metode *System Usability Scale* ini dilakukan dengan menyebarkan *kuesioner* kepada responden. Pengujian ini bertujuan untuk mengetahui penilaian pengguna terhadap sistem yang berhasil dikembangkan (Pawestri, Az-Zahra, and Rusydi 2019). Tes SUS memiliki 10 jenis pertanyaan seperti pada Tabel 2 dan terdapat 5 pilihan jawaban yang telah ditentukan antara lain “Sangat Tidak Setuju”, “Tidak Setuju”, “Ragu-ragu”, “Setuju”, dan “Sangat Setuju”, setiap jawaban memiliki skor 1-5. Pertanyaan bernomor ganjil, skor dihitung dengan cara skor pertanyaan dari pengguna dikurangi 1. Pertanyaan bernomor genap skor akhir didapat dari nilai 5 dikurangi skor pertanyaan yang didapat dari pengguna.

Tabel 2. Pertanyaan SUS

No	Pertanyaan (P)
1	Saya berpikir akan menggunakan sistem ini lagi.
2	Saya merasa sistem ini rumit untuk digunakan.
3	Saya merasa sistem ini mudah digunakan.
4	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya.
6	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi pada sistem ini).
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat.
8	Saya merasa sistem ini membingungkan.
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini.

10	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.
----	---

Penilaian dari responden berhasil dikumpulkan maka selanjutnya dilakukan perhitungan terhadap nilai tersebut guna mendapatkan skor SUS. Gambar 32 memperlihatkan kategori penerimaan skor SUS. Skor akan dianalisa dan diinterpretasikan menggunakan kategori penerimaan (*Acceptability*), skala nilai (*Grade*), dan *Adjectives Ratings* (Astawa, Darmawiguna, and Sugihartini 2019). Minimal skor SUS atau rata-rata yang harus diperoleh sistem agar dapat dikategorikan sebagai sistem yang layak adalah sebesar 70 (Brooke 2013).



Gambar 32. SUS Score

Tabel 3. Pehitungan Skor SUS Responden

Responden	P1	P2	P3	P4	P5	P6	P7	P8	P9	p10	Jumlah	Skor SUS (Jumlah*2,5)
1	3	4	4	3	3	2	2	3	3	3	30	75
2	3	4	2	2	2	2	2	4	2	4	27	67,5
3	4	4	3	2	3	3	4	4	2	3	32	80
4	2	2	3	4	4	3	4	4	2	2	30	75
5	3	3	2	2	2	3	3	3	2	3	26	65
6	2	3	4	2	4	3	3	4	3	4	32	80
7	4	4	3	2	4	3	3	4	4	4	35	87,5
8	4	2	2	2	3	3	3	4	2	2	27	67,5
9	2	3	3	4	2	2	4	3	4	4	31	77,5
10	3	3	4	4	4	2	2	2	3	4	31	77,5
11	2	3	3	2	4	2	4	4	4	4	32	80
12	4	2	2	2	4	3	2	2	4	3	28	70
13	4	4	2	4	3	2	2	2	2	3	28	70
14	3	3	4	4	3	4	4	3	4	4	36	90
15	3	4	3	3	4	3	2	2	4	4	32	80

Tabel 3 merupakan hasil perhitungan skor SUS dari masing-masing responden. Skor SUS responden secara keseluruhan memperoleh nilai rata-rata dari rumus perhitungan skor SUS, jumlah keseluruhan skor SUS dibagi total responden maka di dapatkan hasil 76,16666667. Hasil tersebut tergolong *Acceptable* dengan kisaran rata-

rata 71-100 (Brooke 2013). Skor tersebut mendapatkan nilai *grade scale* C dengan adjective rating *Good* dan masuk dalam kategori penerimaan *Acceptable*.

#### 4. PENUTUP

Berdasarkan hasil penelitian dan pengembangan aplikasi yang telah dibuat di labetta solo, yaitu sistem penjualan ikan cupang berbasis web guna memudahkan penjualan dan pembelian ikan cupang di labetta solo. Penelitian tersebut menggunakan dua pengujian sistem yaitu pengujian *black box* dan pengujian SUS (*System Usability Scale*). Hasil pengujian *black box* yang diperoleh sesuai dengan rencana awal dan hasil pengujian SUS memperoleh nilai rata-rata 76,16 yang dapat dikatakan *Acceptable* dan memperoleh nilai *grade scale* C, dengan *adjective* rating *Good*. Hasil tersebut menunjukkan bahwa sistem layak digunakan.

#### DAFTAR PUSTAKA

- Astawa, I Putu Gina, I Gede Mahendra Darmawiguna, and Nyoman Sugihartini. 2019. "Evaluasi Usability Sistem Informasi Kepegawaian Kabupaten Badung (Simpeg Badung) Menggunakan Metode Usability Testing (Studi Kasus : SMP Negeri 3 Petang)." *Kumpulan Artikel Mahasiswa Pendidikan Teknik Informatika (KARMAPATI)* 8 (2): 209. <https://doi.org/10.23887/karmapati.v8i2.18325>.
- Bahrudin, Endang, Aji Sutomo, and Universitas Tidar Indonesia. 2021. "PEMANFAATAN PLATFORM SOSIAL PADA DAERAH," 66–74.
- Brooke, John. 2013. "SUS : A Retrospective." *JOURNAL OF USABILITY STUDIES* 8 (February): 29–40.
- Cahya, Nilam, and Agung Triayudi. 2021. "JURNAL MEDIA INFORMATIKA BUDIDARMA Implementasi Framework Codeigniter Pada Perancangan Chatbot Interaktif Menerapkan Metode Waterfall." *Jurnal Media Informatika Budidarma* 5 (1): 273–79. <https://doi.org/10.30865/mib.v5i1.2623>.
- Destriana, Rachmat. 2019. "Analisis Dan Perancangan E-Bisnis Dalam Budidaya Dan Penjualan Ikan Cupang Menggunakan Metodologi Overview." *JIKA (Jurnal Informatika)* 3 (1): 51–58. <https://doi.org/10.31000/jika.v3i1.2045>.
- Dwijayanti, Andina, and Puji Pramesti. 2020. "Pemanfaatan Strategi Pemasaran Digital Menggunakan E-Commerce Dalam Mempertahankan Bisnis UMKM Pempek4Beradek Di Masa Pandemi Covid-19." *Ikra-Ith Abdimas* 4 (2): 68–73. <https://journals.upi-yai.ac.id/index.php/IKRAITH-ABDIMAS/article/download/982/772>.
- Essebaa, Imane, and Salima Chantit. 2018. "Tool Support to Automate Transformations from SBVR to UML Use Case Diagram." *ENASE 2018 - Proceedings of the 13th International Conference on Evaluation of Novel Approaches to Software*

- Hwangbo, Hyunwoo, Yang Sok Kim, and Kyung Jin Cha. 2018. “Recommendation System Development for Fashion Retail E-Commerce.” *Electronic Commerce Research and Applications* 28: 94–101. <https://doi.org/10.1016/j.elerap.2018.01.012>.
- Khoironi, Fitria Eviana, and Ida Ayu Nyoman Saskara. 2015. “Analisis Pengaruh Kurs Dollar, Inflasi, Dan Produksi Terhadap Ekspor Ikan Hias Di Provinsi Bali.” *E-Journal EP Unud* 6 (3): 337–61.
- Khomariah, Nuril Esti. 2021. “Implementasi Pemasaran Ikan Hias ‘ Sub Betta Brothers ’ Melalui Social Media Dan Digital Marketing” 2 (1): 196–201. <https://doi.org/10.31949/jb.v2i1.674>.
- Mukti, Akhmad, Muhammad Arief, Luthfiana Sari, Nina Dewi, and Agung Rahayu. 2019. “PERBEDAAN METODE PEMELIHARAAN IKAN HIAS PADA KELOMPOK PEMBUDIDAYA IKAN HIAS DI DESA KARANG SENTUL, KECAMATAN GONDANG WETAN, KABUPATEN PASURUAN, PROPINSI JAWA TIMUR.” *Angewandte Chemie International Edition*, 6(11), 951–952. 10 (1): 11–17.
- Pawestri, Raisadya Hemas, Hanifah Muslimah Az-Zahra, and Alfi Nur Rusydi. 2019. “Evaluasi Usability Aplikasi Mobile Menggunakan Usability Testing Dan System Usability Scale (SUS) (Studi Kasus: SOCO, Althea Dan Sephora).” *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIIK)* 3 (10): 9883–91.
- Priyaungga, Bayu Aji, Dwi Bayu Aji, Mukron Syahroni, Nurul Tri Sukma Aji, and Aries Saifudin. 2020. “Pengujian Black Box Pada Aplikasi Perpustakaan Menggunakan Teknik Equivalence Partitions.” *Jurnal Teknologi Sistem Informasi Dan Aplikasi* 3 (3): 150. <https://doi.org/10.32493/jtsi.v3i3.5343>.
- Putri, Tiara Nabila, Rifnaldi, and Surmiyanti. 2019. “Penggunaan Bahasa Pemrograman PHP Dan MySQL Sebagai Penunjang Sistem Informasi Persediaan Dan Penjualan Secara Online” 5 (2): 61–66.
- Royce, Winston W. 1970. “Managing the Development of Large *Software* Systems (1970).” *Ideas That Created the Future*, no. August: 321–32. <https://doi.org/10.7551/mitpress/12274.003.0035>.
- Sudarmilah, Endah, and Rafiqah Maharani Putri Siregar. 2019. “The Usability of ‘Keepin’ Collect the Trash: Virtual Reality Educational Game in Android Smartphone for Children.” *International Journal of Engineering and Advanced Technology* 8 (4): 944–47.
- Sudarmilah, Endah, Agus Supardi, and Edo Ananda Muliawan. 2012. “Aplikasi Administrasi Laboratorium Pada Rumah Sakit PKU Muhammadiyah Delanggu.” *Jurnal Emitter* 12 (01): 8–15.
- Tran, Lobel Trong Thuy. 2021. “Managing the Effectiveness of E-Commerce Platforms in a Pandemic.” *Journal of Retailing and Consumer Services* 58 (August 2020): 102287. <https://doi.org/10.1016/j.jretconser.2020.102287>.
- Yulistina, Siti Rika, Tika Nurmala, Raden Muhamad Agung Trisakti Supriawan, Sandi Hidayat Ikhsan Juni, and Aries Saifudin. 2020. “Penerapan Teknik Boundary



Value Analysis Untuk Pengujian Aplikasi Penjualan Menggunakan Metode Black Box Testing.” *Jurnal Informatika Universitas Pamulang* 5 (2): 129. <https://doi.org/10.32493/informatika.v5i2.5366>.