

RANCANG BANGUN WEBSITE PENJUALAN TANAMAN DENGAN METODE RAPID APPLICATION DEVELOPMENT

1st Muhammad Hauzan Mufid
faculty of informatics
Telkom University
Banyumas, Indonesia
muhammadhauzanmufid@student.telkomuniversity.ac.id

2nd Muhammad Fajar Sidiq
faculty of informatics
Telkom University
Banyumas, Indonesia
mfsidiq@telkomuniversity.ac.id

3rd Maryona Septiara
faculty of informatics
Telkom University
Banyumas, Indonesia
septiara@telkomuniversity.ac.id

Abstrak — Kemajuan sistem teknologi informasi yang berkembang dengan cepat pada era industri telah mendorong optimalisasi sistem informasi di berbagai bidang, termasuk bisnis tanaman hias. Toko A'AN menghadapi tantangan dalam pengelolaan transaksi, promosi, dan penjualan jarak jauh sehingga menghambat efisiensi operasional. Penelitian ini bertujuan untuk merancang dan membangun sistem penjualan berbasis web yang dapat meningkatkan komunikasi dengan pelanggan, memudahkan pengelolaan data, serta memperluas cakupan pemasaran. Sistem dikembangkan dengan menerapkan metode Rapid Application Development (RAD) dengan pendekatan iteratif dan prototyping, dilengkapi fitur manajemen tanaman, transaksi, dan informasi supplier dengan hak akses berbeda. Proses evaluasi diujikan dengan black box dan System usability scale (SUS) memastikan fungsi sistem sesuai kebutuhan tanpa memeriksa struktur internalnya. Hasil penelitian membuktikan bahwa platform berbasis website ini mampu mengoptimalkan efisiensi operasional Toko A'AN melalui digitalisasi transaksi, optimalisasi promosi, dan pengelolaan informasi pelanggan. Penelitian ini juga berkontribusi dalam pengembangan sistem penjualan berbasis web untuk bisnis tanaman hias.

Kata kunci— *Black Box Testing, Rapid Application, Penjualan, System Usability Scale Toko Tanaman Hias A'AN, Website.*

I. PENDAHULUAN

Pada masa kini, internet telah umum digunakan oleh berbagai kalangan dan terbukti efektif sebagai media pencarian informasi yang relevan. Penggunaannya mudah dan dapat diakses dari berbagai lokasi, termasuk dari rumah. Internet merupakan jaringan luas yang menghubungkan perangkat komputer secara global, memungkinkan pertukaran informasi serta interaksi antar pengguna. Tanaman hias, yang mencakup tanaman berbunga maupun berdaun, tersedia di pasaran dengan beragam bentuk dan warna yang menarik. Beberapa jenis yang populer dan banyak digemari masyarakat saat ini adalah Agrostis, Anthurium, Calathea, Monstera Adansonii, dan Sansevieria.

Setiap orang tentu memiliki preferensi yang berbeda dalam memilih tanaman hias. [1].

Kemajuan teknologi masa kini telah menjadi sarana yang mempermudah aktivitas manusia. Berbagai inovasi teknologi berperan penting dalam perkembangan serta pemrosesan informasi. Masyarakat mulai memanfaatkan teknologi dan sistem penjualan untuk mengakses berbagai informasi serta melakukan transaksi dengan lebih efisien. Perkembangan pesat dalam bidang teknologi informasi dan komunikasi dalam era industri mendorong terciptanya berbagai inovasi guna menyelesaikan berbagai permasalahan. Sistem penjualan hadir sebagai solusi yang mendukung aktivitas harian dengan menyediakan fitur untuk menyimpan, mengolah, dan mengelola data. Selain itu, sistem penjualan juga berperan dalam mengoptimalkan proses bisnis perusahaan. [2].

Toko A'AN merupakan usaha yang berfokus pada penjualan berbagai jenis tanaman hias. Sebagai salah satu bentuk retail tradisional, toko ini beroperasi bersama warung dan pedagang keliling. Meskipun masih tergolong usaha sederhana, Toko A'AN terus berupaya beradaptasi dengan perkembangan teknologi. Namun, dalam menjalankan operasionalnya, toko ini menghadapi sejumlah kendala yang menghambat efisiensi bisnis. Saat ini, transaksi penjualan hanya dapat dilakukan secara langsung di toko, yang membatasi akses pelanggan serta peluang ekspansi pasar. Selain itu, pencatatan transaksi masih dilakukan secara konvensional menggunakan buku, sehingga berisiko mengalami kehilangan, kerusakan, dan kesulitan dalam pengelolaan jangka panjang. Strategi promosi pun masih bergantung pada jaringan kenalan serta aktivitas penjualan keliling saat musim tertentu, yang dinilai kurang efektif dalam menjangkau lebih banyak pelanggan. Toko A'AN mengelola beberapa jenis data penting, termasuk informasi terkait tanaman serta transaksi jual beli. Data tanaman mencakup berbagai jenis tanaman yang tersedia di toko, sedangkan data transaksi berisi catatan rinci mengenai aktivitas penjualan dan pembelian yang terjadi di Toko A'AN.

Sebelum melakukan pengembangan web tersebut, peneliti perlu menentukan siklus pengembangan perangkat lunak atau Software Development Lifecycle (SDLC) yang sesuai. SDLC sendiri dikenal sebagai metode dalam proses pengembangan perangkat lunak. Dalam penelitian ini, istilah SDLC akan merujuk pada metode pengembangan perangkat lunak. Secara umum, metode ini terbagi menjadi dua kategori utama, yaitu metode non-Agile dan metode Agile. Metode non-Agile mencakup pendekatan seperti Waterfall, Iterative, Spiral, dan V-Model. Sementara itu, metode Agile meliputi beberapa pendekatan populer, seperti Scrum, Extreme Programming (XP), Lean, dan Kanban. [3].

Setiap metode pengembangan mempunyai kelebihan dan kekurangan masing-masing. Pilihan metode tergantung pada rincian proyek dan kebutuhan pengembang. Dalam penelitian ini, peneliti menentukan Rapid Application Development (RAD) sebagai metode yang digunakan dalam pengembangan perangkat lunak. Pertimbangan peneliti dalam menentukan metode RAD sebagai metode pengembangan yaitu karena metode ini memungkinkan sistem dapat dikembangkan dengan cepat sehingga dianggap cocok untuk diterapkan dalam pengembangan sistem ini. Metode Rapid Application Development (RAD) menggunakan Metode iteratif (berulang) dalam pengembangan sistemnya. Dalam metode ini, model kerja sistem dirancang pada tahap awal pengembangan dengan tujuan mengidentifikasi kebutuhan pengguna. Biasanya, sistem informasi membutuhkan setidaknya 180 hari untuk dikembangkan, sementara itu, sistem yang dikembangkan dengan metode Rapid Application Development (RAD) dapat diselesaikan dalam kurun waktu 30 hingga 90 hari. [4].

Metode pengujian yang diterapkan dalam sistem penjualan tanaman berbasis web ini adalah black box testing. Metode black box testing merupakan teknik pengujian yang berfokus pada evaluasi fungsionalitas atau kegunaan suatu sistem tanpa memperhatikan struktur internalnya. Pengujian ini juga sangat penting agar bisa menemukan gangguan atau kesalahan dalam suatu sistem informasi perpustakaan berbasis web ini sebelum dirilis secara resmi [5].

Berdasarkan permasalahan yang telah diuraikan sebelumnya, maka penelitian bertujuan untuk merancang sebuah sistem pada Toko Tanaman A'AN yaitu Sebuah platform atau layanan penjualan berbasis web yang dapat memenuhi kebutuhan serta mempermudah operasional toko dalam menjalankan aktivitas bisnis, seperti penjualan, pembelian, promosi, pembayaran, dan transaksi daring, sehingga memungkinkan perluasan jangkauan pemasaran.

II. KAJIAN TEORI

Bagian ini menyajikan serta menguraikan berbagai teori yang berhubungan dengan variabel penelitian. Subjudul dalam bagian ini disusun berdasarkan urutan abjad.

A. Rancang Bangun

Perancangan adalah suatu proses yang bertujuan untuk mengembangkan sistem baru yang mampu menyelesaikan permasalahan pengguna setelah

menentukan sistem yang paling sesuai. Perancangan dan pembangunan merupakan tahapan menerjemahkan hasil analisis ke dalam bentuk perangkat lunak, kemudian mengembangkan atau menyempurnakan sistem yang telah ada, baik secara keseluruhan maupun sebagian. Sementara itu, "bangun" merujuk pada proses implementasi hasil analisis ke dalam bentuk paket perangkat lunak. [6].

B. Website

World Wide Web (WWW), yang sering disebut sebagai "web", merupakan salah satu teknologi utama dalam jaringan internet yang memungkinkan pengguna berinteraksi satu sama lain dan melihat halaman web yang mereka cari. [7].

C. Penjualan

Penjualan merupakan suatu sistem yang mencakup keseluruhan aktivitas bisnis dengan tujuan mencapai target organisasi melalui perencanaan, penetapan harga, promosi, serta distribusi barang, pelayanan, atau ide kepada pasar sasaran. Tujuan utama dari proses ini adalah memenuhi kebutuhan dan keinginan pelanggan sekaligus memperoleh keuntungan dari hasil penjualan. [8].

D. Rapid Application Development (RAD)

Metode ini merupakan metode pengembangan perangkat lunak yang menitikberatkan pada percepatan proses serta partisipasi aktif pengguna. Metode RAD adalah salah satu metode yang digunakan dalam perancangan sistem yang memungkinkan proses pengembangan berlangsung dalam waktu yang relatif cepat. Biasanya diperlukan waktu 180 hari untuk mengembangkan sistem informasi, namun teknik RAD dapat diselesaikan hanya dalam 30 hingga 90 hari. Metodologi pengembangan aplikasi cepat terdiri dari tiga fase untuk merealisasikan sebuah sistem meliputi tahap perencanaan kebutuhan (requirement planning), perancangan model (workshop design), serta penerapan (implementation). Metode ini memungkinkan pengembang untuk menghasilkan sistem dalam waktu singkat melalui penggunaan prototipe dan iterasi cepat. Tahapan utama dalam RAD meliputi:

1. Perencanaan Kebutuhan (*Requirement Planning*)

Selama tahap ini, pemakai dan pengembang dan pengembang berinteraksi secara langsung untuk mengidentifikasi persyaratan yang diperlukan untuk mengatasi masalah yang muncul. Persyaratan yang diidentifikasi harus mampu memecahkan masalah. Hal terpenting Tahap ini melibatkan partisipasi dari kedua pihak yang terlibat. [9].

2. Pemodelan (*Workshop Design*)

Pada titik ini merupakan proses membangun desain prototype dari sistem berdasarkan requirement sebelumnya menggunakan UML (Unified Modeling Language). Dalam tahap ini keaktifan pengguna sangat diperlukan karena menentukan apakah design yang telah dibuat sesuai dengan kebutuhan user atau tidak. Seandainya terdapat hal yang tidak sesuai menurut pengguna maka akan dilakukan perbaikan pada desain [9].

3. Implementasi (*Implementation*)

Setelah tahap pemodelan selesai, dan sudah disetujui oleh pengguna dan pengembang. Pada fase ini, pengembang Mengonversi desain menjadi sebuah program. Setelah program selesai dibuat, tahap berikutnya adalah melakukan pengujian terhadap program tersebut untuk melihat apakah program tersebut layak sebelum diterapkan pada lembaga atau organisasi yang sesuai [9].



Gambar 1

Sumber: Kendall & Kendall dalam (Nurwanti & Meyliana, 2019)

Ilustrasi Model Rapid Application [9].

E. Visual Studio Code

Visual Studio Code merupakan teks editor pemrograman gratis dari yang dapat dijalankan di semua

perangkat desktop. Editor ini memiliki fitur seperti debugging, penyorotan sintaksis, penyelesaian kode otomatis, snippet, refactoring, dan integrasi Git. Pengguna dapat menyesuaikan tema, pintasan keyboard, preferensi, serta menambahkan ekstensi untuk meningkatkan fungsionalitasnya serta menulis lintas platform aplikasi web dan seluler [10].

F. MySQL

MySQL adalah salah satu server basis data populer, SQL adalah bahasa yang digunakan untuk memanipulasi dan mengelola data dalam sebuah basis data., termasuk operasi seleksi dan pemasukan data secara otomatis. Menurut *Van der Lans*, MySQL merupakan sistem manajemen basis data relasional yang menggunakan SQL sebagai bahasa untuk mengelola dan memproses data. [11].

G. Laravel

Laravel adalah framework PHP bersifat open-source yang menerapkan arsitektur Model-View-Controller (MVC). Framework ini dipublikasikan di bawah lisensi MIT dan dikelola melalui platform GitHub untuk berbagi serta mengatur kode sumbernya. [12].

H. Unified Modelling Language (UML)

Teknik pemodelan UML program berorientasi objek (OOP) Adalah standar bahasa pemodelan yang dipakai untuk menggambarkan desain perangkat lunak. Pemodelan adalah penggambaran suatu sistem nyata dalam suatu model ditinjau dari perilaku, struktur fisik, serta karakteristik lain yang menyerupai sistem aslinya. Model yang berkualitas adalah model yang optimal serta dapat diterapkan dalam program komputer. [13].

Dalam Pembuatan sistem ini, penulis akan menggunakan pemodelan diagram UML untuk merancang website yang akan dibuat, termasuk use case diagram, sequence diagram, dan class diagram. Penjelasan lebih lanjut mengenai ketiga diagram tersebut disajikan di bagian berikut.:

1. Use Case Diagram

Menjadi elemen krusial, use case menggambarkan cara individu memanfaatkan dan berinteraksi dengan suatu sistem, sementara aktor merujuk pada entitas yang berhubungan langsung dengan sistem

tersebut. Use case diagram biasanya digunakan untuk mengilustrasikan berbagai aktivitas yang dapat dilakukan oleh pengguna dalam sistem yang dikembangkan. [13].

2. Sequence Diagram

Diagram ini menggambarkan interaksi antar objek dalam sistem berdasarkan urutan kejadian. Aktivasi menunjukkan durasi eksekusi objek, sementara Aktor merepresentasikan peran yang berinteraksi dengan sistem. Collaboration Boundary menggambarkan lingkungan interaksi dan pemantauan objek, dengan garis vertikal paralel sebagai alur proses. Processes mewakili tindakan yang berlangsung bersamaan, Window menampilkan halaman aktif, dan Loop merepresentasikan proses berulang agar sistem berjalan dinamis. Sequence diagram digunakan untuk memahami komunikasi antar komponen dalam sistem [13].

3. Diagram ini merupakan representasi statis dalam UML yang menunjukkan hubungan antar kelas, atribut, metode, dan objek dalam suatu sistem. Digunakan untuk merancang struktur database perangkat lunak, diagram ini mendefinisikan kelas-kelas yang menjadi fondasi sistem. Class Diagram membantu dalam memahami bagaimana komponen sistem saling berhubungan dalam pengembangannya [13].

I. Black Box

Black box testing adalah metode pengujian perangkat lunak yang berfokus pada identifikasi kesalahan dalam sistem aplikasi, seperti tampilan menu yang tidak muncul atau malfungsi pada fitur tertentu. Metode ini mencakup beberapa teknik, di antaranya:

1. Equivalence Partitioning – Membagi input data menjadi beberapa partisi untuk memastikan setiap kategori diuji.
2. Fuzzing – Mendeteksi kesalahan dengan menyuntikkan data cacat dari luar ke dalam sistem.

3. Orthogonal Array Testing – Digunakan ketika jumlah kombinasi input relatif kecil untuk mengoptimalkan pengujian.
4. State Transition Testing – Menguji perubahan kondisi navigasi dan alur sistem menggunakan grafik.
5. Boundary Value Analysis – Mengidentifikasi kemungkinan kesalahan dengan menguji batas minimum dan maksimum input.
6. Cause-Effect Graphing – Menggunakan grafik untuk menganalisis hubungan sebab-akibat dalam pengujian.
7. All Pairs Testing – Menguji semua kemungkinan kombinasi diskrit dari parameter input.

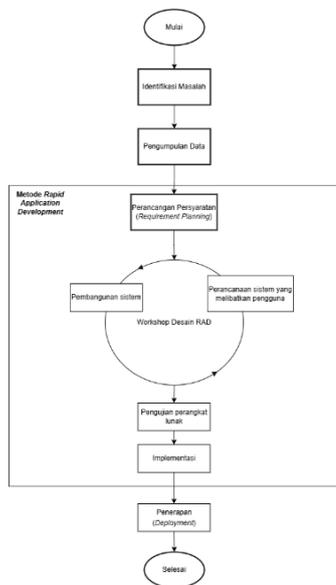
Pengujian dilakukan dengan fokus pada aspek fungsionalitas aplikasi, tampilan antarmuka, serta kesesuaian alur kerja dengan kebutuhan yang telah dirancang oleh pengembang. [14].

J. System Usability Scale (SUS)

Menjadi salah satu metode pengujian SUS merupakan metode evaluasi yang digunakan untuk menilai tingkat kemudahan penggunaan suatu sistem berdasarkan pengalaman subjektif pengguna. Metode ini menggunakan kuesioner berisi 10 pernyataan dengan skala Likert. Skor SUS dihitung dalam rentang 0-100 dan dikategorikan dalam tingkat penerimaan pengguna seperti Not Acceptable, Marginal, atau Acceptable. SUS membantu menilai efektivitas desain antarmuka serta mengidentifikasi aspek yang perlu diperbaiki untuk meningkatkan pengalaman pengguna. [15].

III. METODE

Untuk mendukung penelitian ini, diperlukan suatu diagram alir yang dapat dijadikan pedoman untuk memudahkan penelitian. Gambar 1 menunjukkan diagram alir penelitian ini.



Gambar 2
Diagram Alir Penelitian

A. Identifikasi Masalah

Pada tahap penelitian ini, peneliti menemukan permasalahan yang dihadapi Toko Tanaman A'AN. Observasi dan wawancara di Toko Tanaman A'AN mengungkapkan bahwa penulisan data transaksi Masih diterapkan secara manual/konvensional, sehingga kurang efisien dalam hal waktu dan biaya.

B. Pengumpulan Data

Penelitian ini menggunakan teknik untuk mengumpulkan data yang diperlukan. Adapun metode pengumpulan datanya adalah sebagai berikut:

1. Wawancara dilakukan melalui sesi tanya jawab antara peneliti dan pemilik toko guna memperjelas permasalahan yang ada di Toko Tanaman A'AN.
2. Observasi dilakukan berdasarkan situasi toko dan lokasi Toko Tanaman A'AN.
3. Studi Pustaka Tinjauan pustaka adalah metode Penghimpunan data dilakukan melalui jurnal serta buku yang memiliki relevansi dengan penelitian ini.
4. Dokumentasi adalah cara pengumpulan data hasil observasi dan didokumentasikan sebagai bukti pengumpulan data.

C. Pembangunan Sistem dan Implementasi Metode RAD

Metode pembangunan sistem RAD (Rapid Application Development) digunakan dalam penelitian ini dan tool untuk mendukung proses desain yaitu UML (Unified Modeling Language). Metode RAD cepat beradaptasi, memungkinkan peneliti mempercepat proses desain dan mengumpulkan informasi tentang persyaratan dan kebutuhan melalui umpan balik langsung dari pengguna.

D. Perencanaan Persyaratan (Requirement Planning)

Tahapan yang pertama yaitu tahap perencanaan persyaratan (Requirement Planning) dimana peneliti melakukan pertemuan dengan pengguna untuk mengidentifikasi persyaratan yang diperlukan untuk permasalahan yang ada. Persyaratan yang diidentifikasi harus mampu memecahkan masalah. Setelah

persyaratan diidentifikasi, lanjut ke fase berikutnya yaitu fase workshop design.

E. Workshop Design (Pemodelan)

Tahapan pemodelan (Design Workshop) adalah sebuah tahapan untuk membangun desain sistem berdasarkan requirement planning yang telah ditentukan. Tahap Workshop Design merupakan bagian dari proses iteratif dalam metode Rapid Application Development (RAD). Dalam siklus iterasi ini, terdapat tiga langkah utama yang harus dilalui, yaitu:

a) Perencanaan sistem yang melibatkan pengguna. Pada tahap ini peneliti melakukan perancangan sistem dengan menerapkan UML (Unified Modeling Language), yang meliputi Use Case Diagram, Activity Diagram, Sequence Diagram, serta Class Diagram. Setelah tahap perancangan selesai, pengguna turut serta dalam memberikan umpan balik untuk memastikan bahwa desain sistem telah sesuai dengan kebutuhan mereka.

b) Pada tahap pengembangan sistem, peneliti melakukan pengkodean berdasarkan perancangan yang telah dibuat sebelumnya. Setelah itu, sistem akan melalui tahap pengujian untuk memastikan fungsionalitasnya berjalan sesuai dengan yang diharapkan.

c) Pengujian perangkat lunak Pada tahap ini peneliti melakukan pengujian dengan sosialisasi kepada karyawan Toko Tanaman A'AN. Cara pengujian black box testing dengan mengakses sistem informasi toko. Pengujian dilakukan untuk memastikan fitur serta fungsi yang telah dirancang berjalan sesuai dengan rencana.

F. Implementasi

Tahap implementasi merupakan tahap akhir. Setelah proses iterasi pada workshop design selesai dilakukan, selanjutnya peneliti menyerahkan sistem yang telah selesai dibuat kepada pihak Toko Tanaman A'AN.

G. Penerapan (Deployment)

Tahap deployment merupakan langkah akhir dalam penelitian ini. Pada tahap ini, sistem yang telah selesai dikembangkan dipastikan dapat digunakan di lingkungan nyata oleh pengguna. Langkah-langkah yang dilakukan meliputi instalasi sistem pada perangkat pengguna, pemberian pelatihan kepada pengguna terkait cara pengoperasian sistem, serta monitoring awal Untuk memastikan bahwa setiap fitur berfungsi sesuai dengan kebutuhan serta ekspektasi pengguna.

IV. HASIL DAN PEMBAHASAN

Use Case

A. Perencanaan Kebutuhan

Pada tahapan ini merupakan tahap awal di mana pengguna (user) dan peneliti melakukan pertemuan untuk mengidentifikasi kebutuhan dalam mengembangkan sistem. Pada tahap ini, dilakukan komunikasi untuk mendiskusikan proses yang dapat dijalankan oleh sistem dan aktor dalam website yang dijabarkan sebagai berikut:

1. User/Pelanggan

Pelanggan dapat melakukan pembelian tanaman melalui website dengan cara:

- a) Melakukan registrasi dan login terlebih dahulu.
- b) Memilih kategori tanaman atau mencari tanaman tertentu menggunakan fitur pencarian.
- c) Menambahkan produk ke keranjang belanja.
- d) Melakukan checkout dengan mengisi data alamat pengiriman dan metode pembayaran.
- e) Setelah pembayaran selesai, pelanggan dapat melihat status pesanan pada menu "Pesanan Saya".

2. Admin (Owner)

Admin memiliki berbagai fitur untuk mengelola website, yaitu:

- a) Mengelola Produk: Menambah, mengedit, dan menghapus produk seperti tanaman hias dan bibit.
- b) Mengelola Kategori: Menambah, mengedit, dan menghapus kategori.
- c) Mengelola Pesanan: Melihat daftar pesanan yang masuk, memproses pesanan, dan mengubah status pesanan (misalnya, "Dikirim" atau "Selesai").
- d) Mengelola Pelanggan: Melihat data pelanggan yang telah melakukan registrasi.
- e) Mengelola Laporan Penjualan: Melihat rekapitulasi transaksi yang telah dilakukan dalam bentuk laporan.

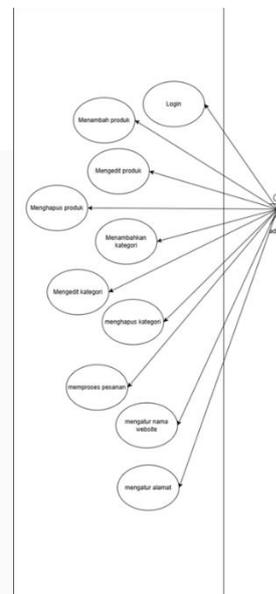
B. Workshop Design

Perencanaan Melibatkan Pengguna

Perencanaan sistem yang akan dikembangkan meliputi Use Case Diagram, Activity Diagram, Sequence Diagram, serta Class Diagram. Rancangan UML tersebut akan diuraikan lebih lanjut pada bagian berikut:

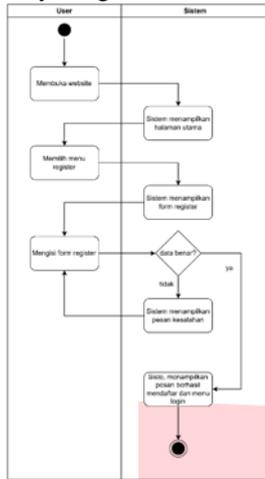


Gambar 3
Use case user



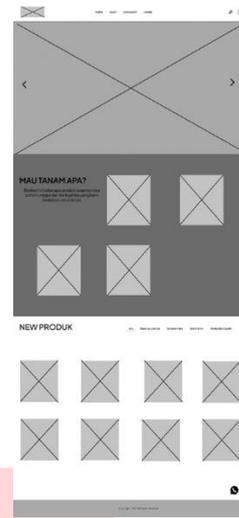
Gambar 4
Use case admin

b. Activity Diagram



Gambar 5

Activity diagram user registrasi



Gambar 8

Wireframe halaman home

f. Pembangunan Sistem

Pada tahap ini, peneliti melakukan pengembangan sistem dengan menerapkan pemrograman menggunakan bahasa PHP serta Framework Laravel. Berikut ini adalah penjelasan mengenai kode yang digunakan:

```

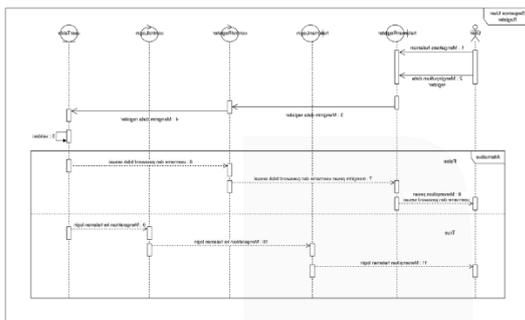
18 class OrderController extends Controller
19 {
20     protected $order;
21     public function __construct($order)
22     {
23         $this->order = $order;
24     }
25     public function index($status = null)
26     {
27         if($status == null){
28             $data['order'] = $this->order->get();
29         }
30         $data['order'] = $this->order->query()->where('status', $status)->get();
31     }
32     return view('backend.feature.order.index', compact('data'));
33 }
34
35 public function show($id)
36 {
37     $data['order'] = $this->order->find($id);
38     return view('backend.feature.order.show', compact('data'));
39 }
40
41 public function update($request)
42 {
43     $request->validate(['status' => 'required']);
44     $data['order'] = $this->order->query()->where('invoice_number', $request->invoice_number)->first()->update($request->only('status', 'receipt_number'));
45     return back()->with('success', __('message.order_update'));
46 }
47
48 }
    
```

Gambar 9

order controller

Kode gambar 9 adalah sebuah controller bernama OrderController yang digunakan untuk mengelola data pesanan dalam sebuah sistem berbasis Laravel. Controller ini memiliki dependensi pada model Order yang diolah menggunakan repository menggunakan CrudRepositories untuk mempermudah operasi CRUD.

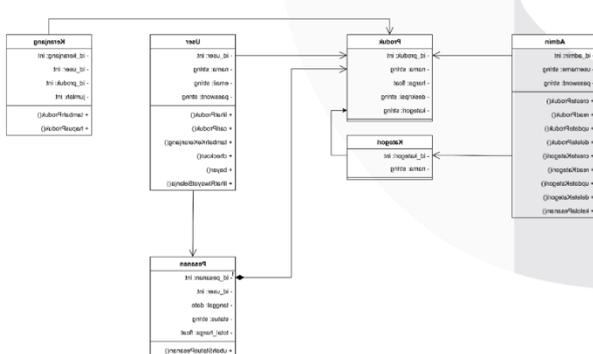
c. Sequence Diagram



Gambar 6

Sequence diagram user-registrasi

d. Class Diagram



Gambar 7

Class Diagram

e. Desain Wireframe

Pada tahap ini peneliti melakukan pembuatan desain wireframe website AAN Plant Shop. Berikut merupakan hasil dari desain wireframe yang telah dibuat :

```

30 class CategoryController extends Controller
31 {
32     protected $category;
33     public function __construct(Category $category)
34     {
35         $this->category = new CrudRepositories($category);
36     }
37
38     public function index()
39     {
40         $data['category'] = $this->category->get();
41         return view('backend.master.category.index', compact('data'));
42     }
43
44     public function create()
45     {
46         return view('backend.master.category.create');
47     }
48
49     public function store(Request $request)
50     {
51         $this->category->store($request->all(), true, ['thumbnail' => 'category']);
52         return redirect()->route('master.category.index')->with('success' => ['message.store']);
53     }
54
55     public function delete($id)
56     {
57         $this->category->hardDelete($id, true, ['thumbnail' => 'category']);
58         return redirect()->back()->with('success' => ['message.delete']);
59     }
60
61     public function edit($id)
62     {
63         $data['category'] = $this->category->find($id);
64         return view('backend.master.category.edit', compact('data'));
65     }
66
67     public function update(Request $request, $id)
68     {
69         if(!isset($request->thumbnail)){
70             $this->category->update($id, $request->all(), true, ['thumbnail' => 'category']);
71         }else{
72             $this->category->update($id, $request->all());
73         }
74         return redirect()->route('master.category.index')->with('success' => ['message.update']);
75     }
76
77     public function show($id)
78     {
79         $data['category'] = $this->category->find($id);
80         return view('backend.master.category.show', compact('data'));
81     }
82 }

```

Gambar 10

CategoryController

Kode gambar 10 adalah sebuah controller bernama CategoryController yang digunakan untuk mengelola data kategori dalam aplikasi berbasis Laravel. Controller ini menggunakan model Category dan repository CrudRepositories untuk mempermudah operasi CRUD.

```

30 class ProductController extends Controller
31 {
32     protected $product;
33     public function __construct(Product $product, Category $category)
34     {
35         $this->product = new CrudRepositories($product);
36         $this->category = new CrudRepositories($category);
37     }
38
39     public function index()
40     {
41         $data['product'] = $this->product->get();
42         return view('backend.master.product.index', compact('data'));
43     }
44
45     public function create()
46     {
47         $data['category'] = $this->category->get();
48         return view('backend.master.product.create', compact('data'));
49     }
50
51     public function store(Request $request)
52     {
53         $data = $request->except('image');
54         $this->product->store($data, true, ['thumbnail' => 'product.thumbnail']);
55         return redirect()->route('master.product.index')->with('success' => ['message.store']);
56     }
57
58     public function show($id)
59     {
60         $data['product'] = $this->product->find($id);
61         return view('backend.master.product.show', compact('data'));
62     }
63
64     public function edit($id)
65     {
66         $data['product'] = $this->product->find($id);
67         $data['category'] = $this->category->get();
68         return view('backend.master.product.edit', compact('data'));
69     }
70
71     public function update(Request $request, $id)
72     {
73         if(!isset($request->thumbnail)){
74             $this->product->update($id, $request->all(), true, ['thumbnail' => 'product.thumbnail']);
75         }else{
76             $this->product->update($id, $request->except('image'));
77         }
78         return redirect()->route('master.product.index')->with('success' => ['message.update']);
79     }
80
81     public function delete($id)
82     {
83         $this->product->hardDelete($id);
84         return redirect()->route('master.product.index')->with('success' => ['message.delete']);
85     }
86 }

```

Gambar 11

ProductController

Kode gambar 11 adalah sebuah controller bernama ProductController yang digunakan untuk mengelola data produk dalam aplikasi Laravel. Controller ini menggunakan repository CrudRepositories untuk mempermudah operasi CRUD pada model Product dan Category.

```

30 class DashboardController extends Controller
31 {
32     protected $chartOrder;
33     protected $chartOrderPie;
34     public function __construct(OrderCart $chartOrder, OrderChartPie $chartOrderPie)
35     {
36         $this->chartOrder = $chartOrder;
37         $this->chartOrderPie = $chartOrderPie;
38     }
39
40     public function index()
41     {
42         $data['total_product'] = Product::count();
43         $data['total_user'] = User::count();
44         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
45         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
46         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
47         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
48         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
49         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
50         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
51         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
52         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
53         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
54         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
55         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
56         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
57         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
58         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
59         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
60         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
61         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
62         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
63         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
64         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
65         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
66         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
67         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
68         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
69         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
70         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
71         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
72         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
73         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
74         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
75         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
76         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
77         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
78         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
79         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
80         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
81         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
82         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
83         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
84         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
85         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
86         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
87         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
88         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
89         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
90         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
91         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
92         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
93         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
94         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
95         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
96         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
97         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
98         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
99         $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();
100        $data['total_pending'] = Order::where('status', '!=', 'selesai')->count();

```

Gambar 12

DashboardController

Kode 12 adalah sebuah DashboardController dalam aplikasi Laravel yang digunakan untuk mengelola tampilan dashboard admin. Controller ini memiliki dua dependency, yaitu OrderCart dan OrderChartPie, yang digunakan untuk membangun data grafik.

C. Pengujian Perangkat Lunak

Pada tahap ini peneliti melakukan pengujian sistem. Pengujian yang dilakukan oleh peneliti yaitu pengujian fungsionalitas menggunakan BlackBox Berikut merupakan hasil pengujian menggunakan BlackBox:

Tabel 1
Pengujian BlackBox

No	Skenario Pengujian	Test Case	Hasil Diharapkan	Hasil Pengujian	Kesimpulan
1	User membuka website	User membuka website di browser	Sistem menampilkan halaman utama	Sistem menampilkan halaman utama	Berhasil
2	User login dengan data valid	User memasukkan username dan password yang valid	Sistem menampilkan halaman dashboard	Sistem menampilkan halaman dashboard	Berhasil
3	User login dengan data tidak valid	User memasukkan username atau password yang salah	Sistem menampilkan pesan error "Username atau password salah"	Sistem menampilkan pesan error "Username atau password salah"	Berhasil
4	User login tanpa mengisi form login	User tidak mengisi username dan password	Sistem menampilkan pesan error "Form login"	Sistem menampilkan pesan error "Form login"	Berhasil

No	Pengujian Fungsionalitas	Total Berhasil	Total Tidak Berhasil
1	Developer	47	0
Total		47	0

Dengan hasil pengujian black box oleh semua responden diatas, maka dapat dilakukan perhitungan dengan teknik analisis deskriptif. Berikut merupakan perhitungan dengan teknik analisis deskriptif :

$$\text{Presentasi Kelayakan} = 127/127 \times 100\% = 100\%$$

Dari hasil perhitungan diatas, nilai 100% untuk pengujian yang berhasil dinilai sangat layak apabila mengacu pada nilai yang tertera pada tabel presentase kelayakan (Tabel 2.2). Berdasarkan hasil tersebut, dapat disimpulkan bahwa website desa margasana dengan fitur pengajuan surat dinilai Sangat Layak

Tabel 2
Penguian SUS

Skor Hasil Hirung										Jumlah	Nilai (Jumlah x 2.5)
Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10		
4	4	4	4	2	4	4	4	4	4	38	95
4	3	4	4	4	4	4	4	4	4	39	98
4	4	4	4	4	4	4	4	4	4	40	100
4	4	4	4	4	4	4	4	4	4	40	100
4	4	4	4	4	4	4	4	4	4	40	100
4	4	4	4	4	4	4	4	4	4	40	100
4	4	4	4	4	4	4	4	4	4	40	100
4	4	4	4	4	4	4	4	4	4	40	100
4	4	4	4	4	4	4	4	2	4	38	95
4	4	4	4	4	4	4	4	3	4	39	98
4	4	4	4	4	4	4	4	1	4	37	95
Skor Rata-rata (Hasil Akhir)											96

Penulisan ini menghasilkan sistem penjualan toko tanaman bibit/hias a'an berbasis web sebagai solusi dari permasalahan yang telah dibahas. Penelitian ini menerapkan metode **Rapid Application Development (RAD)** dalam perancangan sistem, Framework yang digunakan untuk merancang sistem ini yaitu Laravel dengan penerapan MVC sebagai konsep model dari laravel. Sistem ini diuji menggunakan metode Black Box Testing untuk mengevaluasi fungsionalitasnya, serta System Usability Scale (SUS) guna mengukur tingkat kepuasan pengguna. Hasil pengujian menunjukkan skor 96, yang termasuk dalam kategori Excellent, berdasarkan evaluasi Black Box Testing.

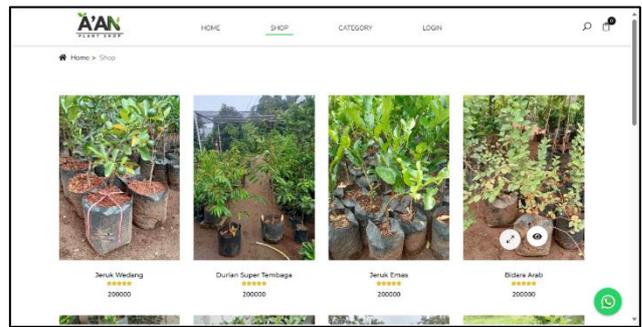
D. Implementasi

Tahap ini merupakan tahap akhir dalam alur metode Rapid Application Development (RAD). Pada tahap ini, penulis memaparkan hasil dari proses pengembangan website A'AN Plant Shop.



Gambar 13
halaman utama

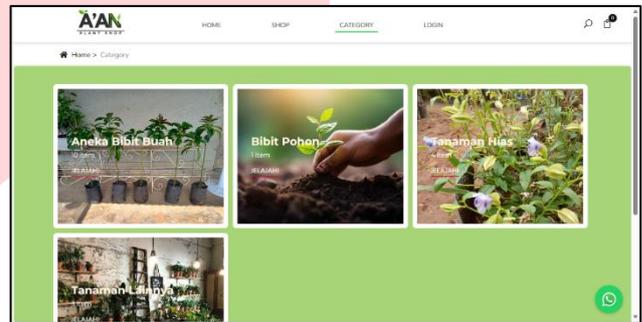
Gambar 13 adalah tampilan gambar landing page/utama website penjualan tanaman hias AAN Plant Shop. Terdapat navbar dan carousel pada halaman ini.



Gambar 14

Halaman shop

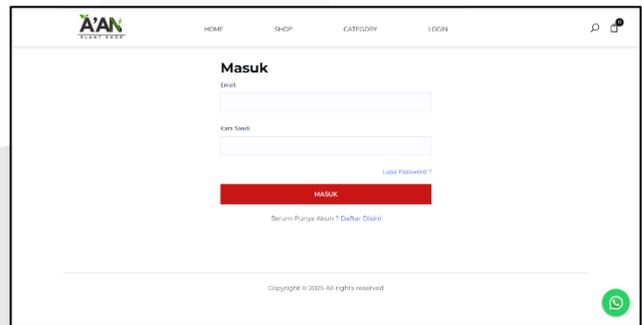
Gambar 14 adalah tampilan gambar page shop website penjualan AAN Plant Shop. Terdapat navbar dan daftar produk.



Gambar 15

Halaman kategori

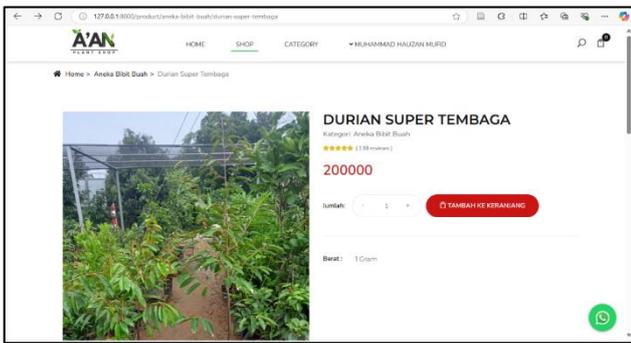
Gambar 15 adalah tampilan gambar page kategori dari website penjualan AAN Plant Shop. Terdapat navbar dan daftar kategori yang dimiliki website AAN Plant Shop.



Gambar 16

Halaman login

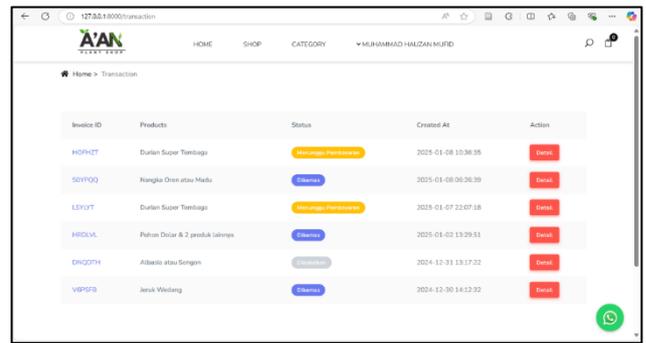
Gambar 16 adalah tampilan gambar page login dari website AAN Plant Shop. Terdapat navbar dan field email dan password serta tombol login.



Gambar 17

Halaman detail produk

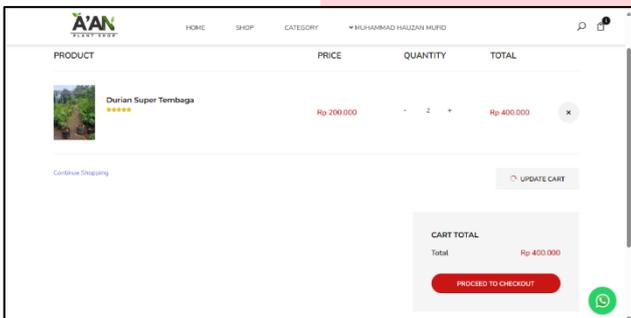
Gambar 17 adalah tampilan gambar page detail produk website AAN Plant Shop. Pada halaman ini terdapat navbar, data produk, dan tombol tambah ke keranjang untuk memasukan gambar ke keranjang.



Gambar 20

Halaman riwayat belanja

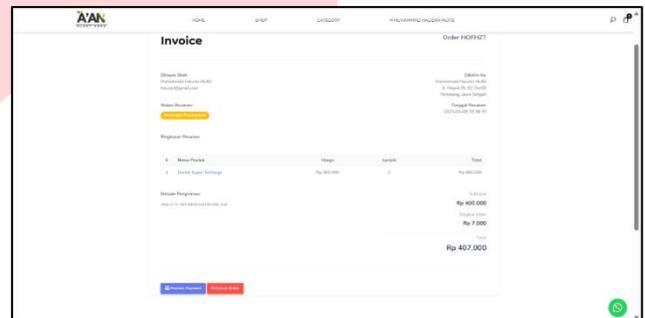
Gambar 20 adalah tampilan gambar page Riwayat belanja dari website AAN Plant Shop. Pada halaman tersebut terdapat navbar dan daftar Riwayat belanja yang telah dilakukan.



Gambar 18

Halaman keranjang

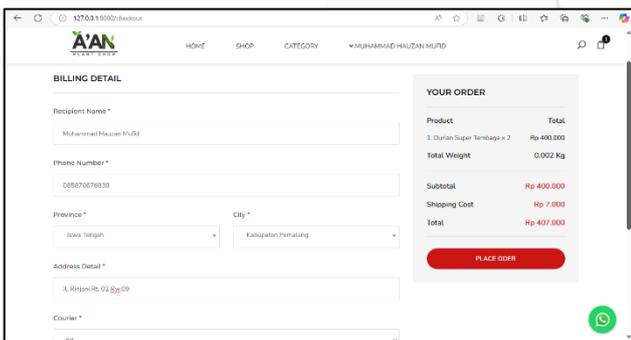
Gambar 18 adalah tampilan gambar page keranjang dari website AAN Plant Shop. Pada halaman ini terdapat navbar, daftar produk di keranjang dan tombol proceed to checkout.



Gambar 21

Halaman invoice

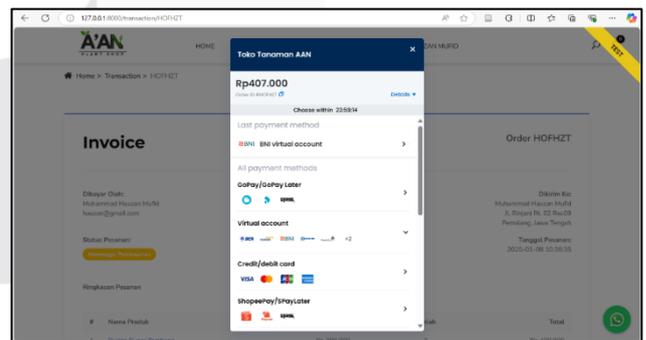
Gambar 21 adalah tampilan gambar page invoice website AAN Plant Shop. Pada halaman tersebut terdapat navbar dan data dari invoice pesanan.



Gambar 19

Halaman checkout

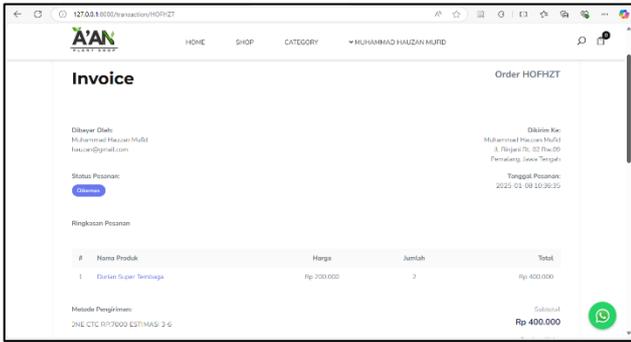
Gambar 19 adalah tampilan gambar page checkout pada website AAn Plant Shop. Pada halaman tersebut terdapat navbar dan kolom pengisian data checkout.



Gambar 22

Halaman pembayaran midtrans

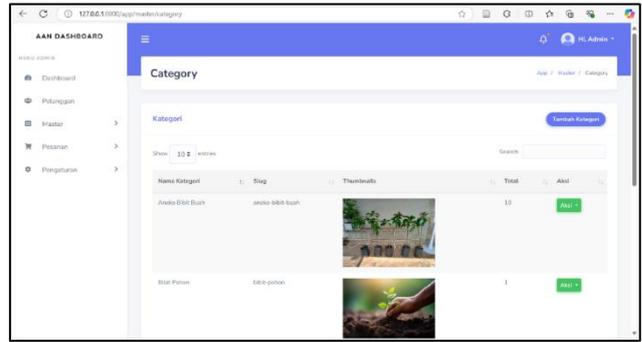
Gambar 22 adalah tampilan gambar page pembayaran melalui midtrans. Pada halaman pembayaran tersebut terdapat beberapa metode pembayaran yang disediakan oleh midtrans.



Gambar 23

Halaman invoice Ketika sudah melakukan pembayaran

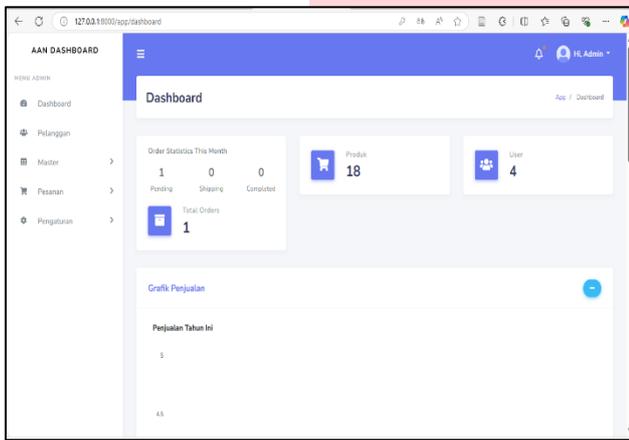
Gambar 23 adalah tampilan gambar page invoice website penjualan AAN Plant Shop. Pada halaman tersebut terdapat navbar dan data pesanan. Terdapat status dikemas ketika sudah melakukan pembayaran



Gambar 26

Halaman kategori

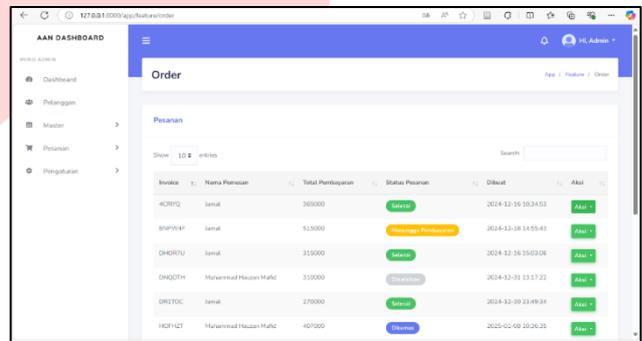
Gambar 26 adalah tampilan gambar page index kategori. Pada halaman tersebut terdapat sidebar dan daftar kategori.



Gambar 24

Halaman admin

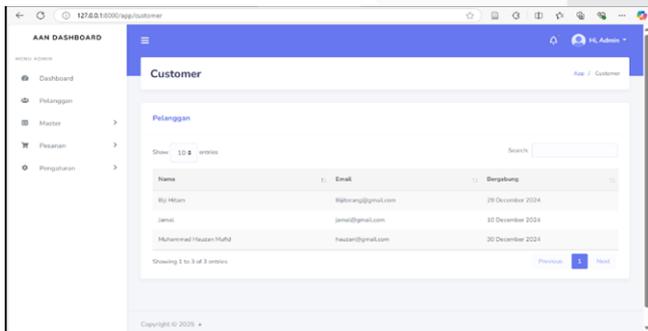
Gambar 24 adalah tampilan gambar page utama admin. Pada halaman tersebut terdapat sidebar, data pesanan, dan grafik penjualan



Gambar 27

Halaman order

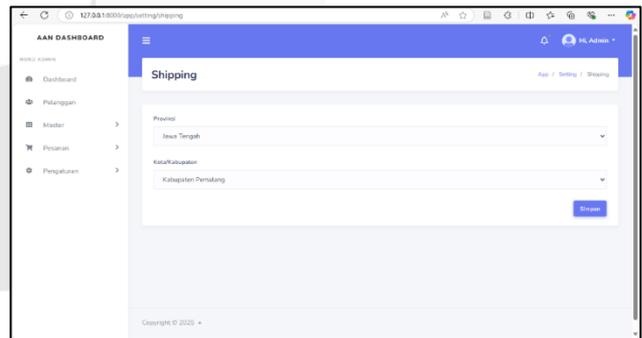
Gambar 27 adalah tampilan gambar page order. Pada halaman tersebut terdapat sidebar dan daftar order yang masuk.



Gambar 25

Halaman menu customer

Gambar 25 adalah tampilan gambar page menu customer. Pada halaman tersebut terdapat sidebar dan daftar customer.



Gambar 29

Halaman pengaturan alamat

Gambar 29 adalah tampilan gambar page pengaturan alamat pada website AAN Plant Shop. Pada halaman tersebut terdapat sidebar, field provinsi, dan field kabupaten yang dapat diubah oleh admin.

E. Penerapan (Deployment)

Tahapan ini merupakan proses deployment, yaitu langkah penting dalam mempublikasikan website agar dapat diakses oleh pengguna melalui internet. Pada

proses ini, website yang telah selesai dikembangkan dipindahkan dari lingkungan pengembangan ke lingkungan produksi. Website tersebut kemudian diunggah ke server hosting yang telah dipilih sesuai dengan kebutuhan proyek. Hosting berfungsi sebagai tempat penyimpanan file dan data yang diperlukan untuk menjalankan website. Selain itu, diperlukan konfigurasi domain agar website memiliki alamat yang mudah diakses oleh pengguna. Domain berfungsi sebagai pengganti alamat IP server, sehingga pengguna dapat mengakses website dengan nama yang mudah diingat. Proses deployment juga melibatkan pengujian untuk memastikan website berjalan dengan baik tanpa adanya error. Dengan langkah ini, website siap digunakan oleh pengguna sesuai dengan tujuan pengembangannya.

V. KESIMPULAN

Hasil penelitian yang telah dilakukan menunjukkan bahwa sistem penjualan online yang dikembangkan berbasis web berhasil meningkatkan proses transaksi yang sebelumnya hanya dilakukan secara langsung, memperluas akses pelanggan, dan menggantikan pencatatan manual dengan penyimpanan data yang terintegrasi. Fitur promosi berbasis web turut meningkatkan efektivitas pemasaran. Pengujian menunjukkan sistem berjalan sesuai kebutuhan pengguna. Disarankan penambahan fitur ulasan pelanggan dan notifikasi stok barang, pemanfaatan strategi pemasaran digital, serta peningkatan keamanan dengan enkripsi data dan autentikasi pengguna. Pelatihan staf serta evaluasi berbasis feedback pengguna perlu dilakukan perlu diperbarui secara berkala agar tetap relevan dan mampu mendukung perkembangan bisnis.

REFERENSI

- [1] M. D. Lusita, Y. I. Chandra, dan E. Orlando, "Penerapan Model Rapid Application Development dalam Membangun Aplikasi E-Commerce Tanaman Hias di Toko Anak Tanam Berbasis Web," *JURNAL INFORMATION SYSTEM*, vol. 2, no. 1, 2022.
- [2] Y. Dwi Wijaya, "Penerapan Metode Rapid Application Development (Rad) Dalam Pengembangan Sistem Informasi Data Toko Penulis Korespondensi." [Daring]. Tersedia pada: <http://www.jurnal.umk.ac.id/sitech>
- [3] A. A. Permana dkk., "Memahami Software Development Life Cycle Penerbit Cv.eureka Media Aksara."
- [4] A. P. Simanungkalit, N. A. Putri, dan V. Tasril, "Rancang Bangun Sistem Informasi Approval Dismantling NTE Telkom Akses dengan Metode RAD (Rapid Application Development)," *INDOTECH Indonesian Journal of Education And Computer Science*, vol. 1, no. 1, 2023.
- [5] M. Nur Ichsanudin, M. Yusuf, S. Jurusan Rekayasa Sistem Komputer, J. Teknik Industri, I. AKPRIND Yogyakarta, dan R. Artikel, "Pengujian Fungsional Perangkat Lunak Sistem Informasi Perpustakaan Dengan Metode Black Box Testing Bagi Pemula Info Artikel Abstrak," vol. 1, no. 2, hlm. 1–8, 2022, doi: 10.55123.
- [6] "R. Sitanggang, T. U. Dachi, dan I. H. G. Manurung, 'Rancang Bangun Sistem Penjualan Tanaman Hias Berbasis Web Menggunakan PHP dan MySQL,' *Jurnal Tekesnos*, vol. 4, no. 1, pp. 84–90, 2022."
- [7] O. Irnawati, G. Bayu, A. Listianto, M. Informatika, dan A. Bsi Bekasi, "Metode Rapid Application Development (RAD) pada Perancangan Website Inventory Pt. Sarana Abadi Makmur Bersama (S.a.m.b) Jakarta 1)." [8] D. Susanti dan A. Tisti Pratiwi, "Sistem Informasi Distributor Penjualan Bibit Tanaman Dan Sayuran Di Kabupaten Majalengka Berbasis Mobile Android," vol. 02, 2020.
- [9] A. Meyliana, L. A. Safitri, dan A. Andriani, "Aplikasi Metode Rapid Application Development (RAD) dalam Perancangan Website PT Sovva Kreasi Indonesia," Online.
- [10] A. Del Sole, *Visual Studio Code Distilled*. Apress, 2019. doi: 10.1007/978-1-4842-4224-7.
- [11] R. Safitri, "Jurnal Tibanndaru Volume 2 Nomor 2, Oktober," 2018.
- [12] C. Gibran, A. R. Dewi, dan E. Hadinata, "Implementasi Framework Laravel Untuk Pengembangan Website Penjualan Ayam Potong Dengan Pemanfaatan Midtrans Menggunakan Metode Fast," *Jurnal Ilmu Komputer dan Sistem Informasi (JIKOMSI)*, vol. 7, no. 1, hlm. 246–253, 2024.
- [13] C. Ayu Binangkit, A. Voutama, dan N. Heryana, "Pemanfaatan Uml (Unified Modeling Language) Dalam Perencanaan Sistem Pengelolaan Sewa Alat Musik Berbasis Website," 2023.
- [14] A. Samdono, A. P. Sari, dan F. P. Aditiawan, "Pengujian Black Box Pada Sistem Informasi Stok Dan Penjualan Berbasis Website Menggunakan Metode Equivalence Partitioning (Studi Kasus: Cv. Algani Karya Mandiri)," 2024.
- [15] S. Sari, A. T. Saadah, D. F. Sugiono, G. D. P. Palunggono, dan M. F. Hidayatullah, "Penerapan Metode System Usability Scale (SUS) pada Pengujian UI/UX Website 'Ternakku.Id,'" *Smart Comp: Jurnalnya Orang Pintar Komputer*, vol. 13, no 2, Apr 2024, doi: 10.30591/smartcomp.v13i2.6275.