

PEMBUATAN SISTEM INFORMASI TOKO SAYURAN BERBASIS WEB MENGGUNAKAN PHP DAN MYSQL

Hermawaty¹, Futri Novi Yanti², Purwadi Budi Santoso³

^{1,2,3}Teknik Informatika Sekolah Tinggi Teknologi Mandala

Emmasantoso1973@gmail.com¹, pn795491@gmail.com²,

purwadiugm87@gmail.com³

ABSTRAK

Dewasa ini perkembangan teknologi informasi sangatlah pesat. Tak terkecuali pada bidang pertanian. Hal ini ditandai dengan terciptanya berbagai alat maupun program yang diperuntukan untuk mempermudah dan memperbaiki hasil pertanian. Orang-orang berlomba menciptakan sayuran dengan kualitas terbaik namun melupakan tujuan utama dari pertanian itu sendiri, yaitu terletak pada bagian penjualan hasil pertanian tersebut. Banyak dari petani yang kesulitan menjual hasil panen terbaik mereka, pada akhirnya mereka harus menjual hasil pertanian tersebut dengan harga yang relatif rendah pada bandar atau pengepul. Dari permasalahan di atas, maka penulis bermaksud memuat aplikasi dimana petani dapat menjual sendiri hasil pertanian mereka secara langsung kepada konsumen. Dimana aplikasi ini sendiri akan di rancang *se-simple* mungkin mengingat aplikasi ini nantinya akan di gunakan oleh petani yang sebagian besar masih awam dengan kemajuan teknologi. Aplikasi ini nantinya akan menggunakan bahasa pemrograman PHP dan MySQL sebagai databasenya.

Kata kunci : Sayuran, WEB, PHP

ABSTRACT

Nowadays, the development of information technology is very rapid. No exception in agriculture. This is marked by the creation of various tools and programs intended to simplify and improve agricultural products. People compete to create the best quality vegetables but forget the main purpose of farming itself, which lies in the selling part of these agricultural products. Many of the farmers have difficulty selling their best crops, in the end they have to sell these agricultural products at relatively low prices to dealers or collectors. From the problems above, the author intends to load an application where farmers can sell their own agricultural products directly to consumers. Where this application itself will be designed as simple as possible considering that this application will later be used by farmers, most of whom are still unfamiliar with technological advances. This application will use the programming language PHP and MySQL as its database.

Keywords: Vegetables, Web, PHP.

1. PENDAHULUAN

Dewasa ini perkembangan teknologi berbasis komputer sangat pesat, karena perkembangan inilah banyak orang yang menggunakan teknologi ini untuk memudahkan pekerjaan atau bisnis mereka, tak hanya instansi saja tetapi banyak rumah (keluarga) yang juga memanfaatkan perkembangan teknologi guna mempermudah pekerjaan sehari-hari. Teknologi sangat dibutuhkan dalam mengimbangi perkembangan zaman juga persaingan didalam perdagangan. Tak terkecuali dalam bidang pertanian, dimana kini banyak para penjual maupun pembeli sayuran yang memanfaatkan teknologi dalam bertransaksi. Seperti kita tahu bahwa pandemi Covid-19 di penghujung tahun 2020 lalu selain membawa dampak buruk juga berdampak pada kemajuan di bidang teknologi. Para ahli IT dan ahli bidang lainnya berlomba untuk menciptakan platform virtual guna menunjang kelangsungan kehidupan meskipun terpisah oleh jarak. Mulai dari trend belanja virtual, bekerja dari rumah, dan tak kalah penting yaitu pada bidang petanian, dimana kini petani dapat menjual sendiri hasil panen mereka tanpa khawatir akan harga pasar yang sering kali tak stabil.

Manfaat utama dari penggunaan aplikasi online shop sayuran adalah kenyamanan dan aksesibilitas. Konsumen dapat membeli sayuran segar dengan mudah tanpa harus meninggalkan rumah. Hal ini sangat menguntungkan bagi mereka yang sibuk atau memiliki keterbatasan waktu dan mobilitas. Selain itu, aplikasi ini juga memungkinkan konsumen untuk memperoleh sayuran segar langsung dari petani atau penjual lokal, sehingga mendukung perekonomian lokal dan menjamin keberlanjutan pertanian. Aplikasi online shop

sayuran juga memberikan keuntungan bagi petani atau penjual. Mereka dapat memperluas jangkauan pasar dan mencapai konsumen yang lebih luas tanpa harus bergantung pada jalur distribusi tradisional. Selain itu, aplikasi ini juga memungkinkan mereka untuk memperoleh harga yang lebih adil dengan menghilangkan perantara atau pengecer. Petani dapat memasarkan produk mereka secara langsung kepada konsumen, sehingga meningkatkan pendapatan dan keuntungan mereka.

2. METODE PERANCANGAN SISTEM

metode yang digunakan dalam pembuatan aplikasi ini yaitu metode *waterfall*. Adapun pengertian SDLC *waterfall* .Menurut (Bassil, 2017) model *waterfall* SDLC (System Development Life Cycle) adalah sebuah metodologi untuk merancang dan membangun sistem perangkat lunak, yaitu proses perancangannya bertahap mengalir semakin ke bawah (mirip dengan air terjun). Adapun model pengembangan sistem adalah *waterfall*.

a. Tahapan Metode Waterfall

Dari pengertian di atas



- mendapatkan tahapan-tahapan metode pengembangan *software* ini. Supaya lebih jelas berikut ini uraiannya.
1. *Requirement*
Pada tahap ini pengembang harus mengetahui seluruh informasi mengenai kebutuhan *software* seperti kegunaan *software* yang diinginkan oleh pengguna dan batasan *software*. Informasi tersebut biasanya diperoleh dari wawancara, survei, ataupun diskusi. Setelah itu informs dianalisis sehingga mendapatkan data-data yang lengkap mengenai kebutuhan pengguna akan *software* yang akan dikembangkan.
 2. *Design*
Tahap selanjutnya yaitu Desain. Desain dilakukan sebelum proses coding dimulai. Ini bertujuan untuk memberikan gambaran lengkap tentang apa yang harus dikerjakan dan bagaimana tampilan dari sebuah sistem yang diinginkan. Sehingga membantu menspesifikasi kebutuhan hardware dan sistem, sistem juga mendefinisikan arsitektur sistem yang akan dibuat secara keseluruhan.
 3. *Implementation*
Proses penulisan code ada di tahap ini. Pembuatan *software* akan dipecah menjadi modul-modul kecil yang nantinya akan digabungkan dalam tahap selanjutnya. Dalam tahap ini juga akan dilakukan pemeriksaan lebih dalam terhadap modul yang sudah dibuat, apakah sudah memenuhi fungsi yang diinginkan atau belum.
 4. *Integration & Testing*
Pada tahap keempat ini akan dilakukan penggabungan modul-modul yang sudah dibuat sebelumnya. Setelah itu akan dilakukan pengujian yang bertujuan untuk mengetahui apakah *software* sudah sesuai desain yang diinginkan dan apakah masih ada kesalahan atau tidak.
 5. *Operation & Maintenance*
Operation & Maintenance adalah tahapan terakhir dari metode pengembangan waterfall. Di sini *software* yang sudah jadi akan dijalankan atau dioperasikan oleh penggunanya.
- b. Kelebihan Metode *Waterfall*
- Berikut ini merupakan beberapa kelebihan yang dimiliki oleh metode waterfall, antara lain:
1. Workflow yang jelas
Dengan menggunakan model SDLC jenis ini, mempunyai rangkaian alur kerja sistem yang jelas dan terukur. Masing – masing tim, memiliki tugas dan tanggung jawab sesuai dengan bidang keahliannya. Serta dapat menyelesaikan pekerjaan sesuai dengan alokasi waktu yang telah ditentukan sebelumnya.
 2. Hasil dokumentasi yang baik

- Waterfall merupakan pendekatan yang sangat metodis, dimana setiap informasi akan tercatat dengan baik dan terdistribusi kepada setiap anggota tim secara cepat dan akurat. Dengan adanya dokumen, maka pekerjaan dari setiap tim akan menjadi lebih mudah, serta mengikuti setiap arahan dari dokumen tersebut.
3. Dapat menghemat biaya
Kelebihan yang selanjutnya tentu saja dari segi *resource* dan biaya yang dikeluarkan oleh suatu perusahaan dengan menggunakan model ini. Jadi, dalam hal ini klien tidak dapat mencampuri urusan dari tim pengembang aplikasi. Sehingga pengeluaran biaya menjadi lebih sedikit. Berbeda dengan metode Agile, yang mana klien dapat memberikan masukan dan feedback kepada tim developer terkait dengan perubahan atau penambahan beberapa fitur. Sehingga perusahaan akan mengeluarkan biaya yang lebih besar daripada Waterfall.
 4. Digunakan untuk pengembangan software berskala besar
Metode ini dinilai sangat cocok untuk menjalankan pembuatan aplikasi berskala besar yang melibatkan banyak sumber daya manusia dan prosedur kerja yang kompleks. Akan tetapi, Model ini juga dapat digunakan untuk proyek berskala kecil dan menengah. Tentu saja disesuaikan dengan kondisi dan kebutuhan proyek yang diambil.
- C. Kelemahan dari Metode Waterfall
Berikut ini terdapat beberapa kelemahan dari metode waterfall, diantaranya adalah sebagai berikut:
1. Membutuhkan tim yang solid
Untuk menggunakan model SDLC ini, tentu saja membutuhkan dukungan dari setiap stakeholders yang ada. Setiap tim harus mempunyai kerja sama dan koordinasi yang baik. Dikarenakan, apabila salah satu tim tidak dapat menjalankan tugas dengan semestinya, maka akan sangat berpengaruh terhadap alur kerja tim yang lain.
 2. Masih kurangnya fleksibilitas
Semua tim dituntut untuk bekerja sesuai dengan arahan dan petunjuk yang telah ditetapkan di awal. Sehingga, klien tidak dapat mengeluarkan pendapat dan feedback kepada tim pengembang. Klien hanya dapat memberikan masukan pada tahap awal perancangan sistem perangkat lunak saja.
 3. Tidak dapat melihat gambaran sistem dengan jelas

Dengan model waterfall, customer tidak dapat melihat gambaran sistem secara jelas. Berbeda dengan model agile yang dapat terlihat dengan baik meskipun masih dalam proses pengembangan.

4. Membutuhkan waktu yang lebih lama

Proses penggerjaan dengan menggunakan waterfall terbilang cukup lama jika dibandingkan dengan model SDLC yang lain. Karena, tahapan penggerjaan aplikasi yang dilakukan satu per satu membuat waktu yang dibutuhkan menjadi lebih lama. Sebagai contoh, tim developer tidak akan bisa melakukan proses coding jika tim designer belum menampilkan tampilan desain dari aplikasi.

2.1. Analisis dan Perancangan

2.1.1. Analisa kebutuhan

Proses bisnis yang dilakukan di Toko Sayur Happy farm dilakukan setiap hari oleh admin happy farm sebagai berikut :

1. Pengecekan stok ketersediaan barang
2. Melakukan riset pasar dan memperbarui harga apabila terdapat kenaikan atau penurunan harga.
3. Update daftar barang setiap hari
4. Melakukan pengecekan pesanan
5. Memperbarui update tentang pesanan setiap ada pesanan masuk
6. Melakukan traksaksi baik secara *transfer* ataupun secara langsung
7. Melakukan pengecekan pembayaran pada setiap pesanan

8. Mengubah status pesanan apabila pesanan telah selesai

3. PEMBUATAN SISTEM

3.1. Kebutuhan fungsional sistem

Tabel 2.1 kebutuhan fungsional

Commented [u1]:

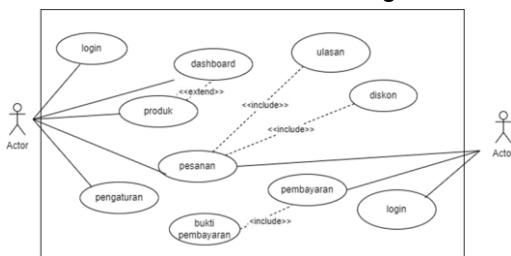
| | |
|--------|-----------------------------|
| KF-001 | Login |
| KF-002 | Menu |
| KF-003 | Menu pengolahan data barang |
| KF-004 | Tambah data barang |
| KF-005 | Edit data barang |
| KF-006 | Hapus data barang |
| KF-007 | Laporan data barang |
| KF-008 | Menu stok barang |
| KF-009 | Tambah stok barang |
| KF-010 | Edit data barang |
| KF-011 | Hapus data barang |
| KF-012 | Menu penjualan |
| KF-013 | Tambah transaksi |
| KF-014 | Edit data barang |
| KF-015 | Hapus transaksi |
| KF-016 | Menu laporan |
| KF-017 | Laporan data barang |
| KF-018 | Laporan stok barang |
| KF-019 | Tambah akun |
| KF-020 | Edit akun |
| KF-021 | Hapus akun |

Tabel 2.2 kebutuhan Nonfungsional

3.6. Usecase Diagram

| | |
|---------|---|
| KNF-001 | Dapat di akses menggunakan internet |
| KNF-002 | Pembeli harus registrasi sebelum membeli barang |
| KNF-003 | Setiap akun dilengkapi dengan kata sandi dan username |
| KNF-004 | Sistem harus dapat memastikan keamanan data user |
| KNF-005 | Memiliki tampilan antarmuka yang mudah di pahami |

Gambar 3.1 usecase diagram



Tabel 3.1 Skenario Login Admin

| | |
|--|---|
| Nama Use Case | Login |
| Aktor | Admin |
| Prekondisi | Aktor akan masuk ke sistem |
| Hasil yang diharapkan | Aktor berhasil masuk ke sistem |
| Aksi Aktor | Reaksi Sistem |
| 1. Membuka sistem | |
| 2. Masuk ke halaman login | |
| 3. Isi kolom <i>username</i> dan <i>password</i> | |
| 4. Klik tombol login | |
| | 5. Memeriksa validasi <i>username</i> dan <i>password</i> |
| | 6. Masuk ke halaman utama sistem |

Tabel 3.2 Skenario Gagal Login Admin

| | |
|--|--|
| Nama Use Case | Gagal Login |
| Aktor | Admin |
| Prekondisi | Aktor akan masuk ke sistem |
| Hasil yang diharapkan | Aktor gagal masuk ke sistem |
| Aksi Aktor | Reaksi Sistem |
| 1. Membuka sistem | |
| 2. Masuk ke halaman login | |
| 3. Isi kolom <i>username</i> dan <i>password</i> | |
| 4. Klik tombol login | |
| | 5. Memeriksa validasi <i>username</i> dan <i>password</i> |
| | 6. Menampilkan pesan <i>username</i> dan <i>password</i> tidak valid |

Tabel 3.3 Skenario Logout Admin

| | |
|------------------------------|-----------------------------------|
| Nama Use Case | Logout |
| Aktor | Admin, Pegawai Gudang |
| Prekondisi | Aktor di dalam sistem |
| Hasil yang diharapkan | Aktor berhasil keluar dari sistem |
| Aksi Aktor | Reaksi Sistem |
| 1. Aktor dalam sistem | |
| 2. Klik menu logout | |
| | 3. Berhasil keluar dari sistem |

Tabel 3.4 Skenario Produk Admin

| | |
|--|---|
| Nama Use Case | Produk |
| Aktor | Admin |
| Prekondisi | Aktor di dalam sistem |
| Hasil yang diharapkan | Sistem akan menampilkan informasi produk yang Admin kelola |
| Aksi Aktor | Reaksi Sistem |
| 1. Aktor masuk kedalam sistem | |
| | 2. sistem menampilkan halaman beranda |
| 3. Aktor akan menekan tombol produk | |
| | 4. System akan menampilkan informasi produk |
| 5. Actor mengubah, menambah, mengurangi, atau menghapus informasi produk | |
| | 6. System akan menyimpan infomasi baru lalu menampilkan informasi produk yang telah di perbarui |

Tabel 3.5 Skenario Pesanan Admin

| | |
|---|---|
| Nama Use Case | Pesanan |
| Aktor | Admin |
| Prekondisi | Aktor di dalam sistem |
| Hasil yang diharapkan | Sistem akan menampilkan pesanan yang di terima |
| Aksi Aktor | Reaksi Sistem |
| 1. Aktor masuk kedalam sistem | |
| | 2. system menampilkan halaman beranda |
| 3. actor masuk kedalam halaman pesanan | |
| | 4. System akan menampilkan laporan pesanan yang di terima |
| 5. Actor segera mengerjakan pesanan | |
| 6. Actor mengonfirmasi pesanan yang telah di kerjakan | |
| | 7. System menampilkan pesanan yang telah selesai |

Tabel 3.6 Skenario Pengaturan Admin

| Nama Use Case | Pengaturan |
|---|--|
| Aktor | Admin |
| Prekondisi | Aktor di dalam sistem |
| Hasil yang diharapkan | Sistem akan menampilkan halaman pengaturan |
| Aksi Aktor | Reaksi Sistem |
| 1. Actor masuk kedalam sistem | |
| | 2. System menampilkan halaman beranda |
| 3. actor masuk kedalam halaman pengaturan | |
| | 4. System akan menampilkan halaman pengaturan |
| 9. Actor mengubah, menambah, mengurangi atau menghapus data pada halaman pengaturan | |
| 10. | |
| | 11. System menampilkan data yang telah di perbarui |

Tabel 3.7 Skenario Pembayaran admin

| Nama Use Case | Pembayaran |
|--|--|
| Aktor | Admin |
| Prekondisi | Aktor di dalam sistem |
| Hasil yang diharapkan | Sistem akan menampilkan data pesanan yang telah dibayar |
| Aksi Aktor | Reaksi Sistem |
| 1. Actor masuk kedalam sistem | |
| | 2. System menampilkan halaman beranda |
| 3. actor masuk kedalam halaman pembayaran | |
| | 4. System akan menampilkan laporan pesanan yang telah dilunasi |
| 12. Actor mengkonfirmasi pembayaran yang telah lunas | |
| | 13. System menampilkan invoice dari pesanan |

Tabel 3.8 Skenario Login Pembeli

| Nama UseCase | Login Pembeli |
|--|---|
| Aktor | pembeli |
| Prekondisi | Aktor telah berada dalam sistem |
| Hasil yang di harapkan | Aktor dapat masuk kedalam akun yang dimilikinya |
| Aksi aktor | Reaksi sistem |
| 1. Actor masuk ke halaman login | |
| | 2. Sistem menampilkan halaman login |
| 3. Actor memasukan username dan password | |
| | 4. Sistem akan memeriksa validasi uname dan pass |
| | 5. Sistem berhasil menampilkan halaman setelah proses login |

Tabel 3.9 Skenario Gagal Login pembeli

| Nama Use Case | Gagal Login pembeli |
|--|--|
| Aktor | Pembeli |
| Prekondisi | Aktor telah berada dalam sistem |
| Hasil yang diharapkan | Aktor gagal masuk ke sistem |
| Aksi Aktor | Reaksi Sistem |
| 1. Membuka sistem | |
| | 2. Masuk ke halaman login |
| 3. Isi kolom <i>username</i> dan <i>password</i> | |
| 4. Klik tombol login | |
| | 5. Memeriksa validasi <i>username</i> dan <i>password</i> |
| | 6. Menampilkan pesan <i>username</i> dan <i>password</i> tidak valid |

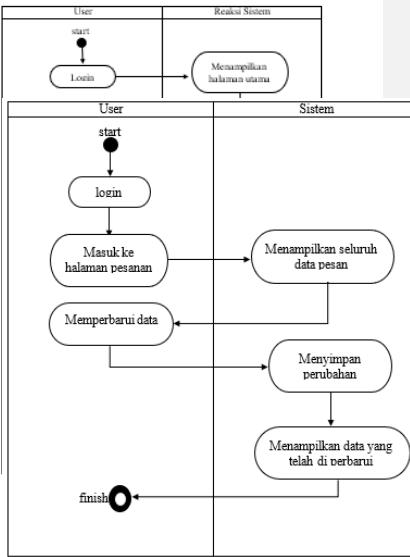
Tabel 3.10 Skenario Registrasi Pembeli

| Nama Use Case | Registrasi Pembeli |
|-------------------------------|---|
| Aktor | Pembeli |
| Prekondisi | Aktor akan membuat akun baru |
| Hasil yang diharapkan | Aktor memiliki akun baru |
| Aksi Aktor | Reaksi Sistem |
| 1. Membuka halaman registrasi | |
| 2. Mengisi form registrasi | |
| | 3. Memeriksa ketersediaan password dan username |
| | 4. Menampilkan konfirmasi bahwa akun telah dibuat |
| | 5. Meminta aktor memasukan kembali password |
| 6. Memasukan password | |
| | 7. Menampilkan beranda sistem |

Tabel 3.11 Skenario Gagal registrasi pembeli

| Nama Use Case | Gagal registrasi |
|--|--|
| Aktor | Pembeli |
| Prekondisi | Aktor akan masuk ke sistem |
| Hasil yang diharapkan | Aktor gagal membuat akun baru |
| Aksi Aktor | Reaksi Sistem |
| 1. Membuka sistem | |
| | 2. Masuk ke halaman register |
| 3. Isi kolom <i>username</i> dan <i>password</i> | |
| 4. Klik tombol register | |
| | 5. Memeriksa ketersediaan <i>username</i> dan <i>password</i> |
| | 6. Menampilkan pesan <i>username</i> dan <i>password</i> tidak valid |

3.3 Activity Diagram



Tabel 3.12 Skenario Pesanan Pembeli

| | |
|-------------------------------|---------------------------------------|
| Nama Use Case | Form Pembeli |
| Aktor | Pembeli |
| Prekondisi | Aktor telah berada di form pesanan |
| Hasil yang diharapkan | Aktor berhasil mengisi form pemesanan |
| Aksi Aktor | Reaksi Sistem |
| 1. Membuka form pemesanan | |
| | 2. Masuk ke halaman pesanan |
| 3. Aktor mengisi form pesanan | |
| 4. Klik tombol pesan | |
| | 5. Memeriksa validasi pesanan |
| | 6. Melanjutkan ke halaman pembayaran. |

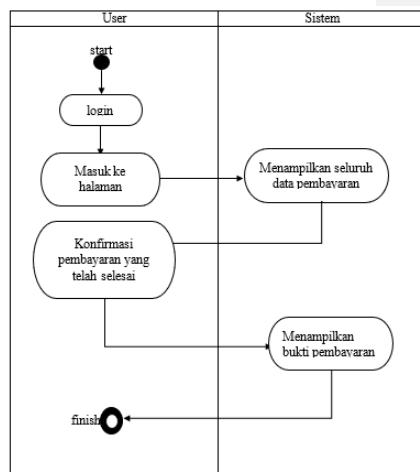
Tabel 3.13 Skenario pembayaran

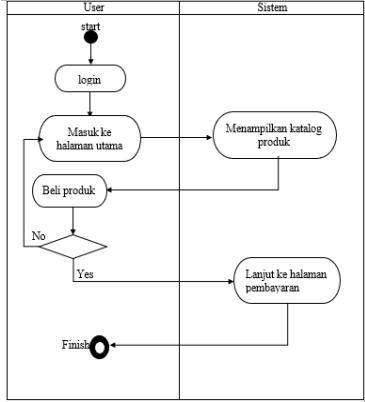
| | |
|----------------------------|------------------------------------|
| Nama Use Case | Form pembayaran |
| Aktor | pembeli |
| Prekondisi | Aktor perada dalam form pembayaran |
| Hasil yang diharapkan | Aktor berhasil membayar pesanan |
| Aksi Aktor | Reaksi Sistem |
| 1. Membuka form pembayaran | |
| | 2. Menampilkan form pembayaran |
| 3. Isi kolom pembayaran | |
| 4. Klik tombol bayar | |
| | 5. Memeriksa validasi pembayaran |
| | 6. Mencetak invoice |

Gambar 3.2 Login Admin

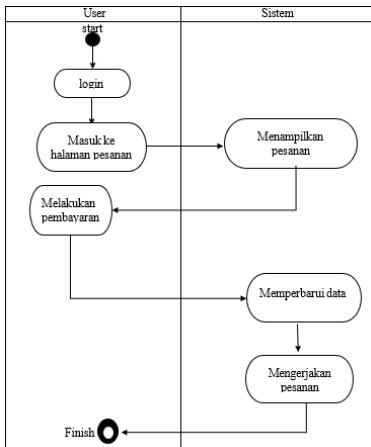
Gambar 3.3 Pesanan Admin

Gambar 3.4 pembayaran admin



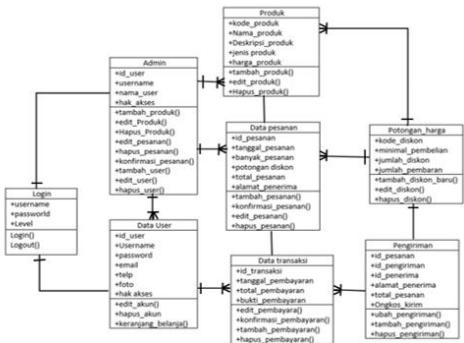


Gambar 3.5 Login Pembeli

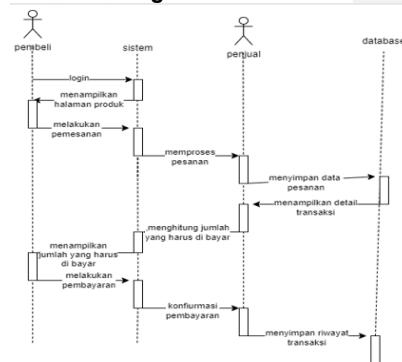


Gambar 3.6 Transaksi Pembeli

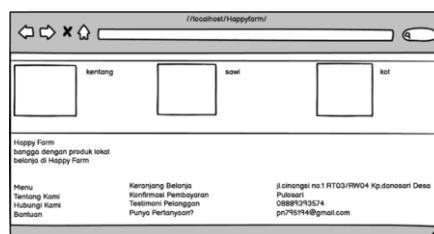
Gambar 3.7 Sequence diagram



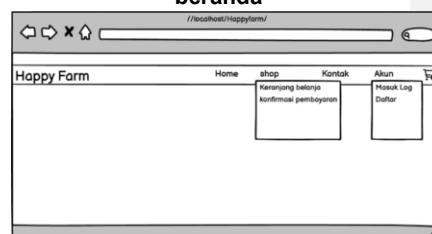
Gambar 3.8 Class diagram



3.4 Perancangan Antarmuka



Gambar 3.9 Halaman beranda



Gambar 3.10 Menu beranda

Gambar 3.11 Dashboard

| kelola pembayaran | | | | |
|-------------------|--------------|---------------|------------|-------------------------------------|
| Nama | Kode pesanan | Total pesanan | Nomor telp | Status |
| Giacomo Guizzoni | #10867 | 200.000 | 0918765332 | <input checked="" type="checkbox"/> |
| Marco Botton | #108770 | 300.000 | 0989977763 | <input type="checkbox"/> |

Gambar 3.12 kelola produk

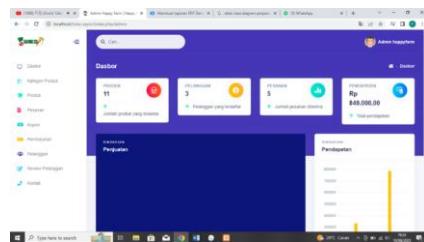
| kelola pelanggan | | | | |
|------------------|--------------|------------|-------------------------------------|-------------------------------------|
| Nama | Kode pesanan | Nomor telp | Status | |
| Giacomo Guizzoni | #10867 | 0918765332 | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| Marco Botton | #108770 | 0989977763 | <input type="checkbox"/> | <input checked="" type="checkbox"/> |

Gambar 3.13 kelola pelanggan

| kelola pesanan | | | | |
|------------------|--------------|-------------|-------------------------------------|-------------------------------------|
| Nama | Kode pesanan | Total bayar | Status | |
| Giacomo Guizzoni | #10867 | 275.000 | <input checked="" type="checkbox"/> | <input type="checkbox"/> |
| Marco Botton | #108770 | 300.000 | <input type="checkbox"/> | <input checked="" type="checkbox"/> |

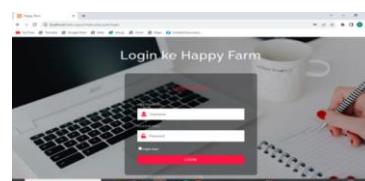
Gambar 3.14 kelola pesanan
Gambar 3.15 kelola pembayaran

3.5 Implementasi Antarmuka



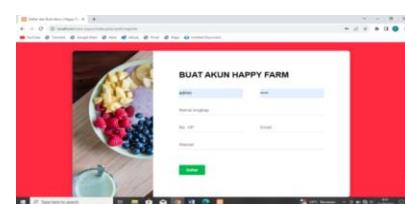
Gambar 3.16 implementasi beranda

Halaman Beranda adalah halaman yang berisikan informasi mengenai produk, dan mengenai toko yang dapat di akses oleh pengguna tanpa harus masuk ke dalam sistem.



Gambar 3.17 Implementasi Login

Halaman login adalah halaman yang dibutuhkan pengguna ketika



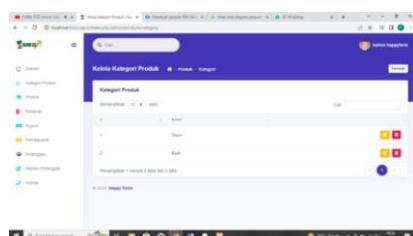
akan masuk kedalam sistem.

Gambar 3.18 Implementasi halaman Registrasi

Halaman registrasi digunakan saat pengguna ingin masuk kedalam sistem namun belum memiliki akun yang terdaftar sebelumnya, maka pengguna akan diarahkan menuju halaman registrasi.

Gambar 3.19 implementasi dashboard

Halaman kategori produk berisikan informasi tentang kategori yang tersedia di Happy Farm, yaitu ada dua kategori buah dan kategori sayur.



Gambar 3.20 Implementasi kategori produk

Halaman ini berisikan informasi mengenai produk yang di jual, bagi admin halaman kelola produk digunakan untuk mengatur serta memperbarui produk yang di tampilkan di beranda setiap harinya.

| No | Kode Order | Nama Pelanggan | Tanggal | Jumlah | Harga | Status |
|----|------------|----------------|--------------------|--------|---------------|---------|
| 1 | AKORD00001 | Happy Farm | Rabu, 27 Juli 2023 | 1 | Rp 70.000,00 | Dikirim |
| 2 | AKORD00002 | Happy Farm | Rabu, 27 Juli 2023 | 1 | Rp 21.000,00 | Dikirim |
| 3 | AKORD00003 | Happy Farm | Rabu, 27 Juli 2023 | 1 | Rp 130.000,00 | Dikirim |
| 4 | AKORD00004 | Happy Farm | Rabu, 27 Juli 2023 | 2 | Rp 140.000,00 | Dikirim |
| 5 | AKORD00005 | Happy Farm | Rabu, 27 Juli 2023 | 1 | Rp 60.000,00 | Dikirim |

Gambar 3.21 implementasi kelola produk

Halaman ini berisikan informasi pesanan masuk yang harus segera di tindak lanjuti untuk kemudian di proses dan dikirim kepada pembeli.

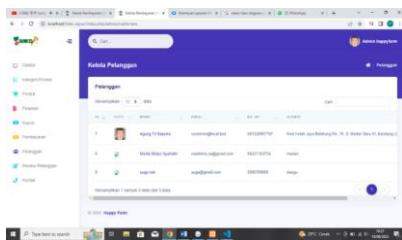
Gambar 3.22 Implementasi Kelola pesanan

Halaman ini berisikan informasi pesanan masuk yang harus segera di tindak lanjuti untuk kemudian di proses dan dikirim kepada pembeli.

| No | Kode Pesanan | Nama Pelanggan | Tanggal | Jumlah | Harga | Status |
|----|--------------|----------------|--------------------|--------|---------------|---------|
| 1 | AKORD00001 | Happy Farm | Rabu, 27 Juli 2023 | 1 | Rp 40.000,00 | Dikirim |
| 2 | AKORD00002 | Happy Farm | Rabu, 27 Juli 2023 | 1 | Rp 100.000,00 | Dikirim |

Gambar 3.23 implementasi pembayaran

Halaman ini berisikan informasi mengenai transaksi yang sudah dibayarmaupun belum terjadi pembayaran, lalu admin akan mengonfirmasi pembayaran apabila pembayaran telah selesai dilakukan oleh pembeli.



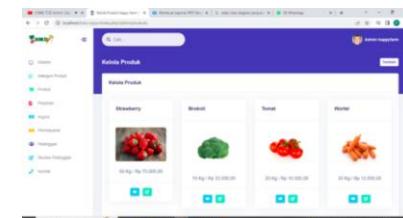
Gambar 2.24 Implementasi kelola pelanggan

Halaman Ini berisikan Informasi data mengenai pembeli yang telah melakukan pemesanan, serta review pelayan yang diberikan pelanggan kepada toko Happy Farm.

3.6 Pengujian Sistem

Metode dalam pengujian sistem aplikasi stok barang ini menggunakan metode Black Box Testing. Dengan memberikan masukan pada sistem dan melihat hasil yang dikeluarkan oleh sistem dengan memperhatikan antarmuka sistem. Hasil yang diharapkan dari pengujian sistem adalah ketetapan sistem dalam

mengelola masuk dan keluarnya untuk mempermudah pengelola data



toko sayur happy farm.

Tabel 3.14 Pengujian Blackbox

| Nama fungsi | Aktor peng uji | Bentuk pengujian | Hasil yang di inginkan | Hasil pengujian | status |
|-----------------------|----------------|-----------------------|---|--|----------|
| Login | Admin | Username & password | Admin/pembeli dapat masuk kedalam sistem | User berhasil masuk kedalam sistem | Berhasil |
| Registrasi | Admin | data user | Pengguna baru dapat membuat akun untuk berbelanja | Pengguna mendapatkan akun untuk berbelanja | Berhasil |
| Tambah | Admin | Tambah produk | Admin dapat menambahkan produk baru di toko | Admin berhasil menambahkan produk baru | Berhasil |
| Edit | Admin | Edit produk | Admin dapat mengubah produk | Admin berhasil mengubah produk | Berhasil |
| Hapus | Admin | Hapus produk | Admin dapat menghapus produk | Admin berhasil menghapus produk | Berhasil |
| Menambahkan beranda | Admin | Menambahkan beranda | User dapat masuk ke halaman beranda | User berhasil masuk ke halaman beranda | Berhasil |
| Melakukan transaksi | Admin | Halaman transaksi | Pembeli dapat melakukan transaksi pembelian | Pembeli berhasil melakukan transaksi pembelian | Berhasil |
| Melakukan pembayaran | Admin | Halaman pembayaran | Pembeli dapat melakukan pembayaran | Pembeli berhasil melakukan pembayaran | Berhasil |
| Konfirmasi pembayaran | Admin | Konfirmasi pembayaran | Admin dapat mengonfirmasi pembayaran masuk | Admin berhasil mengonfirmasi pembayaran masuk | Berhasil |

| Nama fungsi | Aktor peng uji | Bentuk pengujian | Hasil yang di inginkan | Hasil pengujian | status |
|----------------------------|----------------|--------------------|--|--|----------|
| | | | an yang masuk | | |
| Konfirmasi pesanan selesai | Admin | Konfirmasi pesanan | Admin dapat mengubah status pesanan ketika selesai | Admin berhasil mengubah status pesanan | berhasil |
| Review pelanggan | Admin | Komentar | Pembeli dapat menulis ulasan ketika selesai berbelanja | Pembeli berhasil memberikan ulasan | berhasil |

Aplikasi ini memungkinkan pelanggan untuk memilih antara pengiriman

4. sayuran ke alamat mereka atau mengambil langsung di toko. Fitur ini memberikan fleksibilitas kepada pelanggan sesuai dengan preferensi masing-masing.

Dalam era di mana teknologi semakin berkembang, aplikasi toko sayuran berbasis teknik informatika ini merupakan langkah yang tepat dalam mengoptimalkan bisnis pertanian dan perdagangan sayuran.

Dengan memberikan solusi efisien dalam pengelolaan inventori, transaksi digital yang aman, serta fitur-fitur modern lainnya, aplikasi ini berpotensi membawa dampak positif bagi pedagang dan pelanggan dalam industri sayuran.

b. Saran

Adapun saran-saran yang dapat disampaikan untuk pengembangan aplikasi yaitu aplikasi belum sepenuhnya maksimal dalam pembuatan dan perlu dikembangkan lagi dalam pembuatan aplikasi seperti mengelola Happy Farm. Penulis menyadari masih banyak kekurangan dalam pembuatan aplikasi toko sayuran berbasis PHP, maka dari itu penulis mengharapkan kritik dan saran yang membangun untuk perbaikan dikemudian yang akan datang.

4. KESIMPULAN DAN SARAN

a. Kesimpulan

Dari penjelasan di atas maka disimpulkan beberapa keunggulan menggunakan toko sayuran berbasis web, diantaranya :

1. Pengelolaan Inventori yang Lebih Efisien:
Aplikasi ini memberikan fitur pengelolaan inventori untuk melacak stok sayuran secara real-time. Hal ini membantu mengurangi risiko kehabisan persediaan dan mengoptimalkan pengadaan barang.
2. Transaksi Digital yang Aman dan Praktis:
Dengan adanya fitur transaksi digital, pelanggan dapat melakukan pembelian sayuran secara online melalui aplikasi. Keamanan transaksi dijamin melalui protokol keamanan yang ketat, sehingga meningkatkan kepercayaan pelanggan terhadap platform.
3. Pesan Antar dan Pengambilan di Toko:

5. DAFTAR PUSTAKA

1. Agus Saputra. 2012. Membangun Sistem Aplikasi E-commerce & SMS. PT Alexmedia Komputindo.
2. Dicoding. 2022. Apa itu UML :
<https://www.dicoding.com/blog/apa-itu-uml/> (diakses Pada 03 mei 2023)
3. Dicoding. 2022. Apa itu activity diagram :
<https://www.dicoding.com/blog/apa-itu-activity-diagram/> (diakses pada ; 03 mei 2023)
4. Dicoding. 2021. Apa itu usecase :
<https://www.dicoding.com/blog/apa-itu-usecase/> (diakses pada 03 mei 2023)
5. Desa Pulosari.2023. Data desa pulosari :
<https://pulosari.desa.id/> (diakses pada 9 juni 2023)
6. Gamela Id. 2022. Apa itu SLDC Waterfall :
<https://www.gamelab.id/news/1345-mengenal-metode-sdlc-dalam-pengembangan-software> (diakses Pada : 03 mei 2023)
7. Gregorius.2000. Membuat homepage interaktif. Jakarta Alex media komputindo
8. Hidayatullah dkk.2015. pemrograman WEB.Bandung informatika.
9. Jurnal IPSIKOM. Vol. 8 No.1, Juni 2020. ISSN : 2338-4093, E-ISSN : 2686-6382
10. Muhlisunamal. 2021. Pendekatan Perancangan Sistem :
<https://muhlisunamal.blogspot.com/2017/04/pendekatan-perancangansistem.html> (diakses pada 22 juni 2023)
11. Murya, Yosef. 2016. *Framework PHP Yii2 Develop Aplikasi Web Dengan Cepat dan Mudah*. Jakarta: Jasakom
12. Masrur.2016. Pemrograman web dinamis menggunakan java server pages dengan database relasional MySQL.ANDI.
13. MADCOMS.2016.Pemrograman PHP & MySQL untuk pemula.Andi offset.
14. Nurul. 2023. Proyek Akhir Nurul "Aplikasi Stok Barang untuk Xherabb Snack Menggunakan PHP dan MySQL"
15. Nasution dan Prasetyawan.2008.Perencanaan & pengendalian produksi. Graha ilmu.
16. Ridha,Najamudin.2018.Client s Side Scripting dan Server Side Scripting. Tersedia di :
<https://komputerkampus.com/artikel/29/client-side-scriptingdanserver-side-scripting/> (diakses pada tanggal 22 juni 2023.)
17. Rumahweb.com.2021.Apa itu CodeIgniter. Tersedia di:
<https://blog.rumahweb.com/apa-itu-codeigniter-adalah/> (Diakses pada tanggal 22 juni 2023)
18. Thabit, Nabel Amir., 2011. Pencitraan 3 Dimensi Berbasis JavaScript Object Notation (JSON) Pada Hyper Text Markup Language 5 (HTML5).Bandung: Program Studi Ilmu Komputer, Universitas Pendidikan Indonesia.