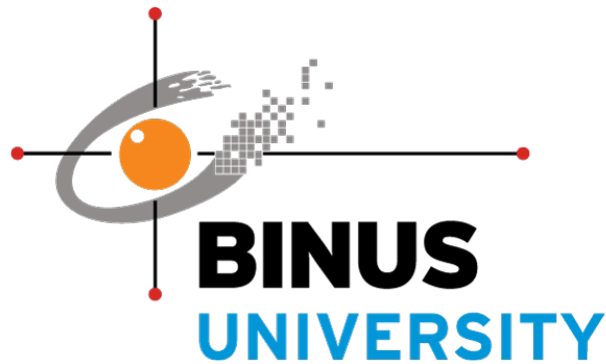


Dokumentasi Web Programming Kelompok 3



Anggota Kelompok:

1. 2702306465 - Darryl Owen Thorian
2. 2702399405 - Kenrick Stephen Chen
3. 2702254404 - Ngurah Ade Januar Aribawa
4. 2702221223 - Rhevell Herliman Senewe

Dosen Pengajar:

Reinert Y. Rumagit, S.Kom., M.TI

Web Programming
Semester Ganjil 2025/2026

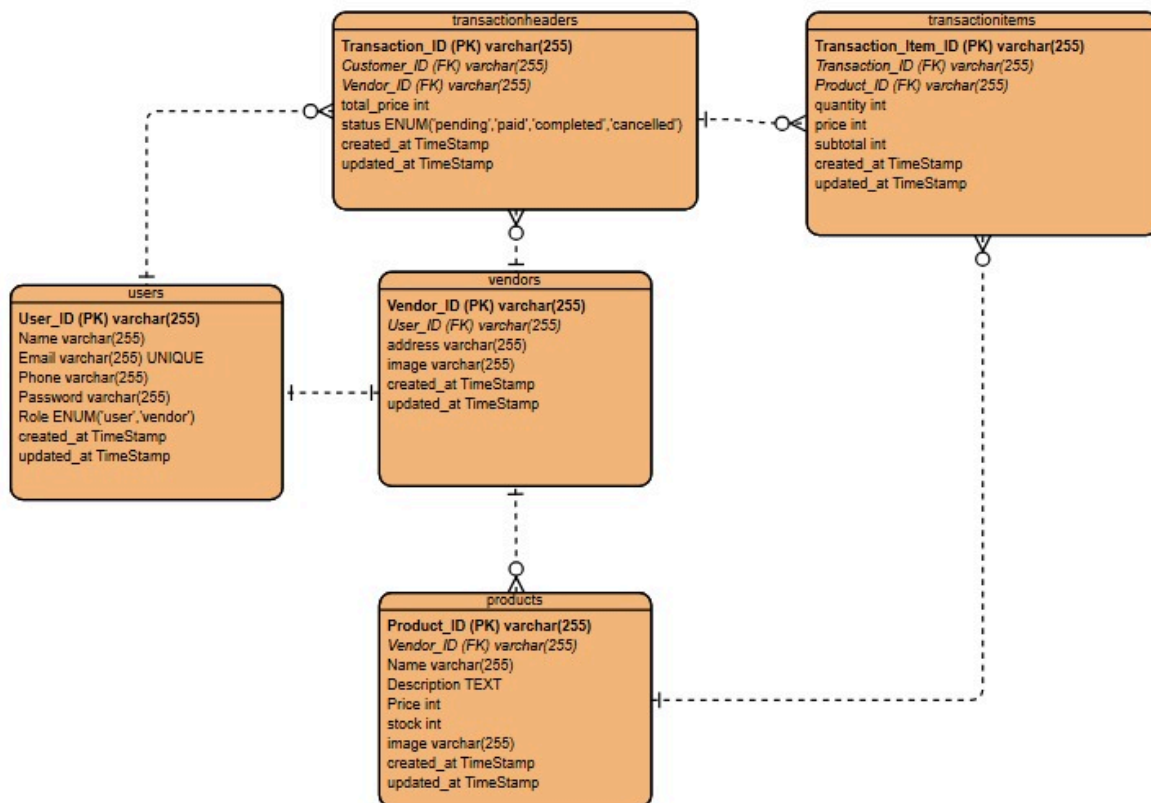
HealthyPick

I. Introduksi

HealthyPick adalah aplikasi berupa web untuk melakukan pesanan atas hidangan-hidangan yang sehat. Dalam website HealthyPick, pelanggan dapat melakukan pemesanan untuk makanan dari vendor-vendor yang telah disiapkan. Website bersifat simpel dan juga terlokalisasi untuk bahasa Indonesia dan Inggris.

- **Entity Relationship Diagram (ERD)**

Entity Relationship Diagram (ERD) adalah representasi visual untuk perancangan struktur berbasis data, ditunjukkan oleh entitas (table) dan atribut data beserta dengan tipenya. Database yang digunakan HealthyPick dapat dilihat perancangannya dengan ERD dibawah.



Gambar 1.1 ERD HealthyPick

Akun pengguna website disiapkan 2 tabel yaitu **users** dan **vendors**; tabel **users** digunakan untuk menyimpan informasi pelanggan sedangkan tabel **vendors** digunakan untuk menyimpan informasi vendor yang menjual produk-produk sehat. Setiap vendor tentu akan menyediakan produk-produk yang unik bagi mereka, maka tabel **products** disiapkan untuk menyimpan informasi produk yang disediakan setiap vendor. Setelah itu, Transaksi yang dilakukan oleh pelanggan akan diandalkan oleh tabel **transactionheaders** dan **transactionitems**. Tabel **transactionheaders** menyimpan

data-data umum seperti siapa yang beli, dengan vendor siapa yang menjual, total harga, dan status pembayaran dari pelanggan; tabel `transactionitems` menyimpan data lebih detail dari sebuah `transactionheader`, seperti detail dari sebuah belanja dimana `transactionheader` adalah kuitansi belanja.

- **Relationships Between Tables**

- **users** → **vendors**: one to many
- **users** → **transactionheaders**: one to one
- **vendors** → **products**: one to many
- **vendors** → **transactionheaders**: one to many
- **transactionheaders** → **transactionitems**: one to many
- **products** → **transactionitems**: one to many

- **Version Control**

Version Control merupakan sebuah sistem yang melacak dan mengelola file yang diupload untuk mengefisiensikan perbandingan dan perkembangan yang paralel. Kami melakukan penyimpanan dan dokumentasi perkembangan dari website kami menggunakan Version Control System yang terkenal bernama GitHub.

- **GitHub**

GitHub memungkinkan pembuatan, penyimpanan, dan pengelolaan kode dengan menggunakan Git sebagai version controlnya. GitHub mempunyai fitur untuk melakukan push files dan branching. Push files digunakan untuk upload sebuah versi agar pengembang dapat memulihkan versi yang lebih lama; branching adalah tempat kerja terpisah untuk membuat perubahan tanpa mengganggu versi main terpenting, jika terkonfirmasi telah bekerja dengan benar maka dapat melakukan merging untuk menggabungkan branch ke dalam main version. Jika ingin mengakses GitHub repository `HealthyPick`, dapat dilakukan melalui file berikut.

<https://github.com/Kephenrick/HealthyPick>

II. Fitur Utama

- **Pengguna User**

Saat pelanggan pertama kali membuka website, akan di redirect pada login page untuk user seperti gambar berikut. Pelanggan tidak dapat melakukan fitur-fitur web sebelum melakukan login.

HealthyPick
Products
Vendors
History
About Us

Login

Email Address

Password

Login

Don't have an account? [Register here](#)
[Login as vendor](#)


© 2026 HealthyPick
EN
ID

Gambar 2.1 Login Page - User


Jika belum memiliki akun maka pelanggan dapat melakukan registrasi akun terlebih dahulu dengan menekan hyperlink dengan teks “Register here” pada gambar tersebut. Setelah itu, pelanggan di redirect pada homepage HealthyPick. Homepage memiliki informasi-informasi umum dari HealthyPick. Pelanggan dapat menelusuri vendor beserta dengan produknya pada halaman vendor yang dapat ditekan lewat navbar di atas.

HealthyPick
Products
Vendors
History
About Us
Logout

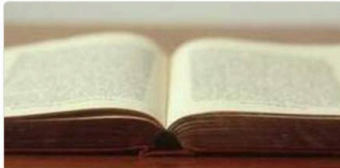
Food Vendors



Braun, Hackett and Hartmann
Address: 539 McGlynn Islands Suite 950
Denesikshire, HI 02944-3438



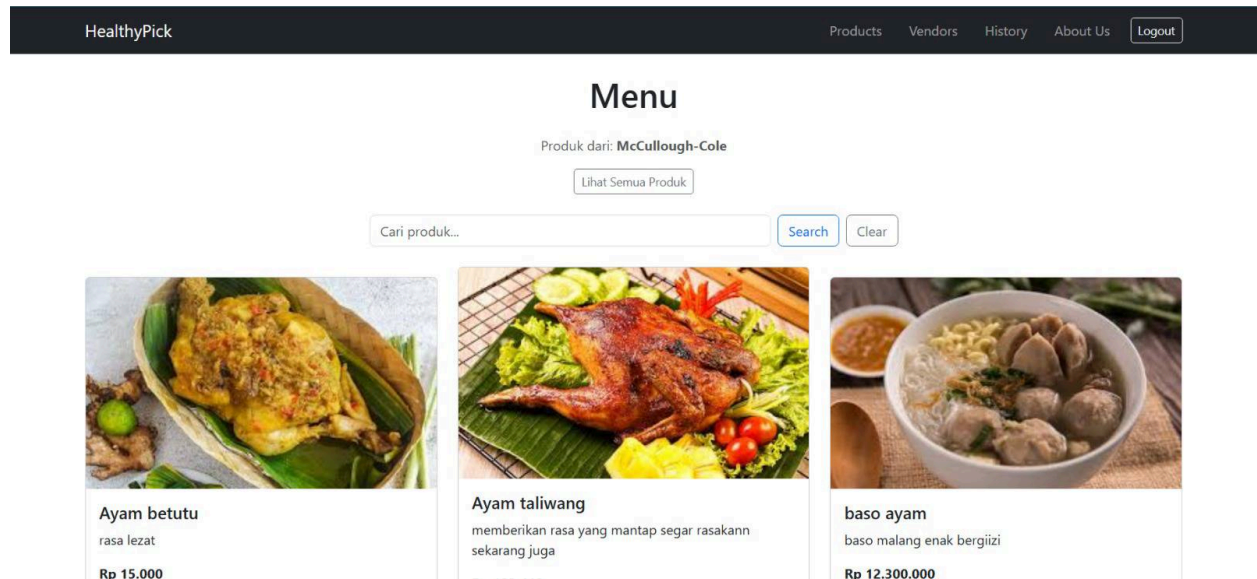
Haley, Kovacek and Welch
Address: 56644 Wisozk Dale Apt. 605 South Diana,
UT 37447-7100



Heathcote-Dietrich
Address: 94940 Kovacek Tunnel Apt. 010 South
Dawnland, WI 05711-0568

Gambar 2.2 Vendor Page - User

Halaman vendor memiliki informasi vendor yang tersedia seperti nama vendor dan alamat vendor. Pengguna dapat menekan button dengan teks “Products” pada body card vendor untuk menelusuri produk-produk yang disiapkan oleh vendor tersebut.



Gambar 2.3 Vendor Product Page - User

Halaman setelah menekan tombol “Products” seperti gambar di atas. Halaman Products ini memberi aneka hidangan dari vendor dalam bentuk menu; informasi makanan yang disediakan berupa: nama hidangan, deskripsi hidangan, dan harga hidangan. Produk tersebut dapat juga ditambahkan ke dalam shopping cart. Contoh tampilan menu jika hidangan sedang kosong akan memiliki tulisan “Out of Stock” pada tombol tambah ke shopping cart.

Payment		Order ID: TXN-1767516751
Order Summary Product: Ayam betutu Quantity: 1 Price per item: Rp 15.000 Amount: Rp 15.000 Order ID: TXN-1767516751	Payment Method Select a payment method <div> <input type="radio"/> Credit / Debit Card Pay with your Visa / MasterCard </div> <div> <input type="radio"/> Bank Transfer Transfer to our bank account </div> <div> <input type="radio"/> E-Wallet Use GoPay/OVO/Dana </div> <div> <input type="radio"/> Cash on Delivery Pay when the order is delivered </div> <div> <input type="button" value="Pay"/> </div>	

Gambar 2.4 Cart Page - User

Setelah selesai menelusuri dan melakukan pembelian, pelanggan dapat melakukan pembayaran di halaman shopping cart. Pelanggan dapat melakukan pemilihan payment method seperti gambar di atas. Setelah menekan tombol “Pay”, transaksi akan diwujudkan pada halaman transaksi.

Transaction

Transaction ID	Product	Quantity	Price per Item	Total Price	Status	Date
00a89ada-7746-4738-854d-6fb9584ab0c3	Ayam betutu	2	Rp 15.000	Rp 30.000	Paid	04 Jan 2026 08:25
ab371e5e-6fc3-48e7-ba6e-e4afedf1a460	Ayam betutu	7	Rp 15.000	Rp 105.000	Pending	04 Jan 2026 08:19

Gambar 2.5 Transaction Page - User

Jika sudah selesai dan berhasil, maka pelanggan dapat mengecek status pembayaran di halaman transaksi. Seperti gambar di atas, pelanggan dapat melihat informasi-informasi untuk produk yang telah mereka pesan beserta dengan statusnya. Pelanggan diharuskan untuk menunggu konfirmasi dari vendor sebelum status akan dinyatakan selesai.

- Pengguna Vendor**
Vendor mempunyai halaman (dashboard) tersendiri di mana, sama seperti pelanggan, vendor harus melakukan log in terlebih dahulu.

Login As Vendor

Email Address

Masukkan email Anda

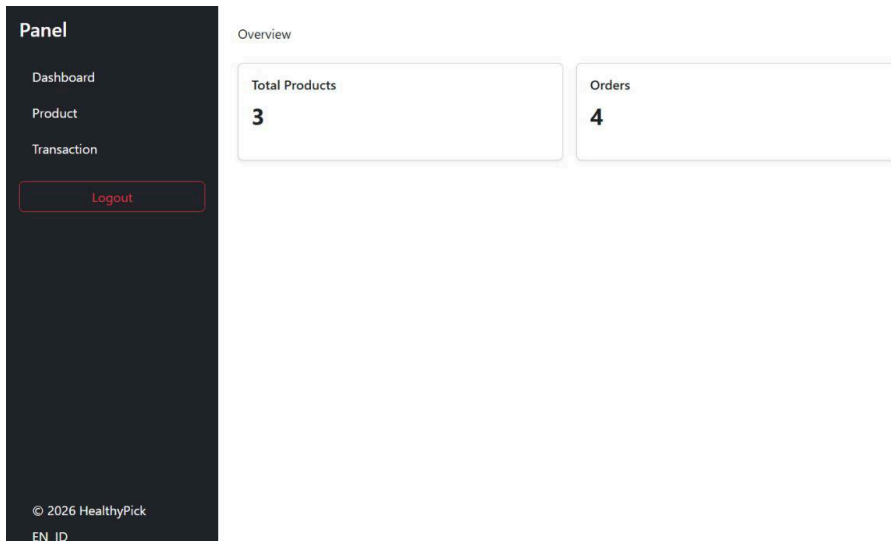
Password

Masukkan password Anda

Login

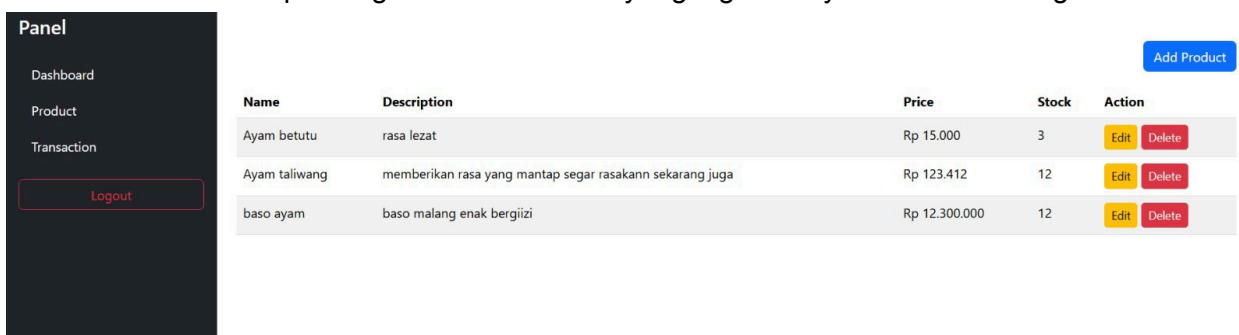
Gambar 3.1 Login Page - Vendor

Vendor akan diminta untuk mengisi email dan password, jika data tersimpan dalam database maka vendor dipersilakan lanjut kepada dashboard.



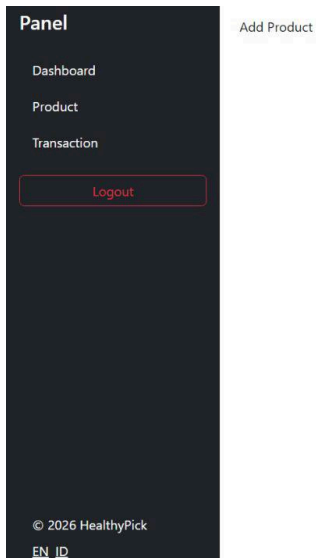
Gambar 3.2 Home Page - Vendor

Setelah melakukan login, vendor dapat melihat overview untuk produk dan berapa total pesanan yang mereka telah melakukan. Dashboard vendor dibuat dengan simpel, vendor ada 3 halaman penting yaitu: Dashboard (Home), Product, dan Transaction. Terakhir ada opsi “Logout” untuk vendor yang ingin menyelesaikan sesi login.



Gambar 3.3 Product Page - Vendor

Saat menekan “Produk” pada sidebar, vendor akan dialihkan ke halaman product. Pada halaman ini tersedia tabel yang mengisikan: nama produk, deskripsi produk, harga produk, stok produk, dan aksi untuk manipulasi data. Pada halaman ini ada 2 fitur penting, yaitu: Add Product dan Edit.

This is a screenshot of a web form titled "Add Product" with a blue header. The form contains several input fields: "Name" with the value "ayam betu", "Description" with the value "ayam lezat bergizi enak", "Price" with the value "15000", and "Stock" with the value "12". Below these is a file upload section labeled "messages.image" with a "Choose File" button and the filename "ayam.jpg". A blue "Add Product" button is at the bottom of the form.

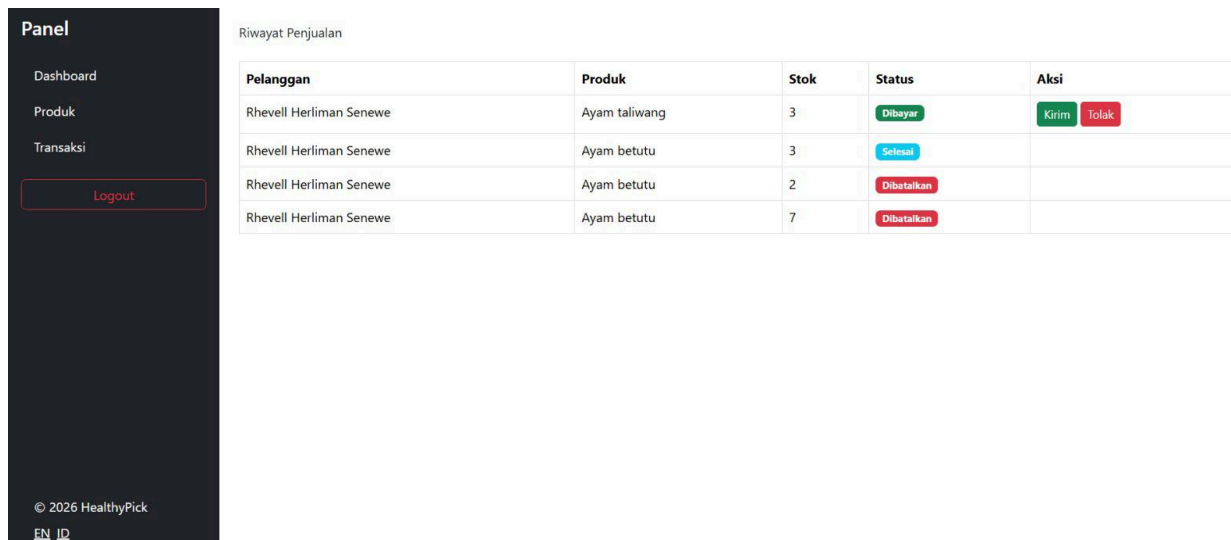
Gambar 3.4 Add Product Page - Vendor

Saat vendor menekan tombol “Add Product”, vendor akan dialihkan ke form untuk menambahkan produk baru. Vendor dipersilakan untuk mengisi: nama, deskripsi, harga, stok, dan gambar untuk produk. Setelah menambahkan produk, produk baru akan ditampilkan kepada pelanggan.

This is a screenshot of a web form for updating a product. It has the same layout as the "Add Product" form but with different values: "Name" is "Ayam betutu", "Description" is "rasa lezat", "Price" is "15000", and "Stock" is "0". The "messages.image" section shows a "Choose File" button and the text "No file chosen". Below this is a "Current Image" section displaying a photo of a cooked chicken dish. A blue "Edit" button is at the bottom.

Gambar 3.5 Update Product Page - Vendor

Jika vendor menekan tombol “Edit” pada tabel header Action, maka vendor akan dialihkan pada form untuk melakukan perubahan pada informasi produk. Selain itu juga tersedia Action “Delete” pada tabel di halaman produk sebelumnya dimana produk akan dihilangkan dari database jika aksi tersebut dilakukan.



Gambar 3.6 Transaction Page - Vendor

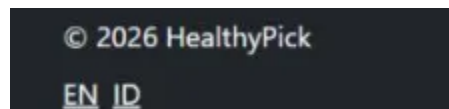
Terakhir adalah halaman transaksi vendor dimana semua pesanan pelanggan diriwayatkan. Pada halaman ini vendor dapat melihat status untuk pesanan pelanggan. Jika pesanan sudah dibayar pelanggan namun belum selesai, maka vendor harus melakukan konfirmasi untuk mengirim atau membatalkan pesanan pelanggan. Saat sudah dilakukan hal tersebut, pelanggan akan melihat update status pada halaman riwayat transaksi mereka.

- **Localization**

Website HealthyPick mempunyai fitur lokalisasi untuk 2 bahasa berbeda, English dan Bahasa Indonesia. Untuk melakukan pengubahan bahasa, pengguna website hanya harus menekan tombol di footer (untuk pelanggan) atau sidebar (untuk vendor) yang tertulis “EN” atau “ID”, dapat dilihat pada gambar di bawah.



Gambar 4.1 Footer - User



Gambar 4.2 Sidebar Bottom - Vendor

III. Panduan Penggunaan

- **Vercel**

Vercel adalah sebuah cloud platform yang digunakan untuk build, deploy, dan scale aplikasi web modern. Vercel terkoneksi langsung pada akun GitHub pengguna, maka penggunaan Vercel memberi hasil yang cepat untuk deployment. Vercel mendukung beberapa frameworks seperti: React, Astro, dan Laravel. Karena Vercel mendukung framework Laravel, kami menggunakan Vercel untuk deploy aplikasi web kami. Domain aplikasi dapat diakses melalui link berikut.

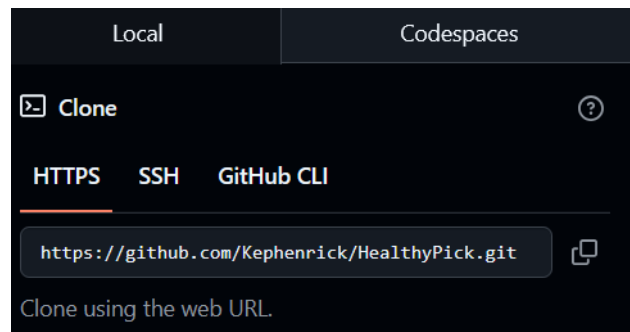
<https://healthypick.vercel.app/user/>

Jika cloud hosting untuk website HealthyPick sedang mengalami masalah, pengguna dapat lanjut ke sub-topic selanjutnya “Manual”. Pengguna akan melakukan instalasi website dari GitHub secara langsung dan konfigurasi untuk website secara mandiri.

- **Manual**

- **Cloning**

Sebelum dapat digunakan, repository harus di clone terlebih dahulu. Cloning dapat dilakukan melalui dua opsi: HTTP atau CLI. Melakukan Git Cloning sangat sederhana, dalam aplikasi Git masuk ke dalam directory yang ingin dipakai lalu dengan menggunakan “git clone” yang dilanjutkan dengan repository url pada gambar dibawah, repository akan selesai diunggah ke local device.



Gambar 5.1 Repository URL

- **Instalasi**

Sebelum website dapat dibuka, harus melakukan instalasi .env file dan vendor folder secara manual terlebih dahulu. Dua file tersebut tidak disimpan dalam GitHub karena bersifat file yang sensitif atau memboros tempat.

- **.env file**

Untuk instalasi .env file caranya sederhana, karena telah disiapkan file .env.example oleh Laravel. Pertama, duplicate terlebih dahulu .env.example dan ubahlah namanya menjadi .env saja (jika langsung diubah tidak masalah namun lebih baik mempunyai backup). Setelah itu dapat melakukan konfigurasi .env file untuk disesuaikan database dan local hostnya.

- **Dependencies**

Dependencies adalah bagian dari Laravel berupa third-party dependencies yang menjalankan framework Laravel agar framework dapat berjalan dengan sesuai. Dependencies Laravel disimpan dalam folder vendor yang tidak di push ke GitHub karena lumayan redundan. Cara instalasi ulang hanya dengan run command “composer install” pada terminal. Laravel akan melakukan instalasi dependencies sesuai dengan requirements yang tertera pada package.json.

- **Autoload**

Autoload adalah mekanisme yang menjalankan PHP class, interface, dll. secara otomatis agar tidak diharuskan untuk melakukan require dan include secara manual. Untuk memastikan file autoload tidak bermasalah, jalankan “php artisan key:generate”.

- **Konfigurasi**

Setelah instalasi selesai dilakukan, pengguna dapat melakukan konfigurasi untuk membuat database dan mempopulasikan database yang akan digunakan oleh website.

- **Migration**

Pembuatan database ke dalam DBMS yang digunakan sekedar menggunakan satu built-in command dari artisan. Sebelum itu, pastikan terlebih dahulu pada .env file bahwa DBMS yang ingin digunakan sesuai dengan konfigurasi pada file tersebut. Jika sudah sesuai maka dapat menjalankan command pada command prompt yang bertulis “php artisan migrate:fresh”.

- **Seeder & Faker**

Setelah melakukan migration untuk tabel ke DBMS, dapat dilihat bahwa entitas dalam database masih kosong semua. Oleh karena itu ada built-in command untuk menjalankan Seeder yang telah kami sediakan untuk tabel users and vendors. Cara menjalankan Seeder mirip dengan migration. Pada command prompt, jalankan command “php artisan db:seed”. Command tersebut akan menjalankan semua Seeder yang disediakan, tetapi jika ingin melakukan secara individu lakukan “php artisan db:seed --class=[SeederFileName]”. Seeder yang kami sediakan mempunyai nama file “ProductSeeder” dan “VendorSeeder”.

Setelah melakukan setup dan instalasi tersebut secara benar, website sudah dapat digunakan. Jika ingin membuka dashboard vendor, maka pada url hanya melakukan penambahan “/vendor” di akhir url tersebut.