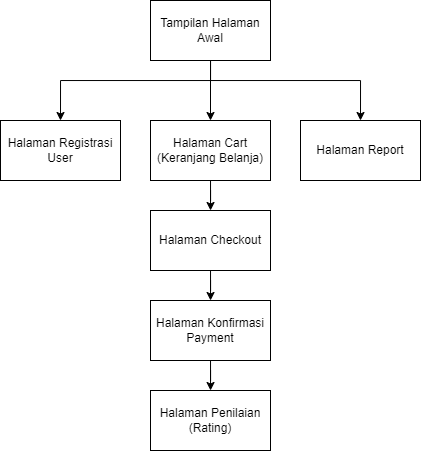
**BAB III**

**DESAIN SISTEM**

Desain adalah suatu perencanaan atau perancangan yang dilakukan sebelum pembuatan sebuah sistem. Pembuatan program membutuhkan desain agar pembuatannya dapat menjadi lebih cepat dan efisien. Pada bab ini akan dijelaskan mengenai desain arsitektural, desain *database*, serta desain interface. Desain ERD, DFD, tampilan tata letak beserta komponennya, serta desain-desain lainnya akan dijelaskan lebih rinci pada bab ini.

1. **Desain Arsitektural**

Dalam subbab ini akan dijelaskan mengenai desain arsitektural pada sistem website marketplace boardgame kami. Desain ini digunakan untuk memudahkan pembuatan sistem dengan adanya pembagian fitur-fitur. Masing-masing modul akan dijelaskan secara lengkap termasuk dengan fungsi-fungsinya.



**Gambar 3. 1**

**Sitemap Website Marketplace Boardgame**

Saat pertama kali membuka website, pengguna akan diarahkan ke halaman awal. Halaman awal akan berisi berbagai produk (boardgame) yang direkomendasikan, serta produk yang sedang trending. Pada halaman utama juga disediakan navigation bar yang berisi berbagai kategori dari produk (boardgame).

Namun sebelum berbelanja, pengguna harus membuat user terlebih dahulu pada website kami. Pengguna dapat membuat user pada halaman registrasi. Setelah membuat user, pengguna akan diminta untuk mengisi alamat pengiriman.

Apabila pengguna sudah menentukan produk yang ingin dibeli, pengguna bisa menambahkan produk tersebut kedalam keranjang. Lalu pengguna bisa menuju halaman cart (keranjang belanja) untuk melihat daftar barang yang telah ditambahkan.

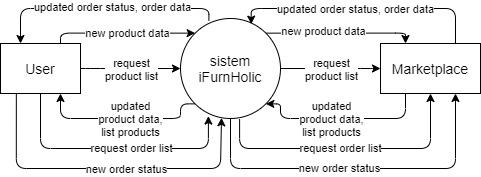
Setelah memilih produk mana saja yang akan dibayar, pelanggan dapat menekan tombol CHECKOUT untuk beralih ke halaman checkout.

Pada halaman checkout akan diberikan informasi mengenai total harga dari seluruh barang yang dibeli, dan pilihan jasa kurir yang tersedia.

1. **Data Modeling**

Dalam subbab ini akan dijelaskan mengenai proses perancangan arsitektur sistem website integrasi marketplace iFurnHolic. Tujuan pembuatan desain ini adalah untuk mempermudah pekerjaan pembuat aplikasi dalam mengembangkan aplikasi. Desain sistem membantu pembuat aplikasi dalam menggambarkan keseluruhan suatu sistem dalam sebuah diagram agar pembuat aplikasi mengerti sistem penting apa saja yang perlu dibuat dan fitur-fitur apa saja yang perlu difokuskan. Pembuatan desain sistem pada website integrasi marketplace iFurnHolic menggunakan Data Flow Diagram atau yang lebih dikenal dengan singkatan DFD.

DFD untuk website integrasi marketplace iFurnHolic dibagi menjadi dua level yaitu Context Diagram dan DFD Level 0. DFD terdiri dari beberapa komponen, seperti pengguna/*terminator*, *process*, data *store*, dan data *flow*. DFD ini menjelaskan beberapa fitur penting yang terdapat pada website integrasi marketplace iFurnHolic. Fitur-fitur penting yang digambarkan pada DFD akan dibedakan sesuai tipe pengguna.

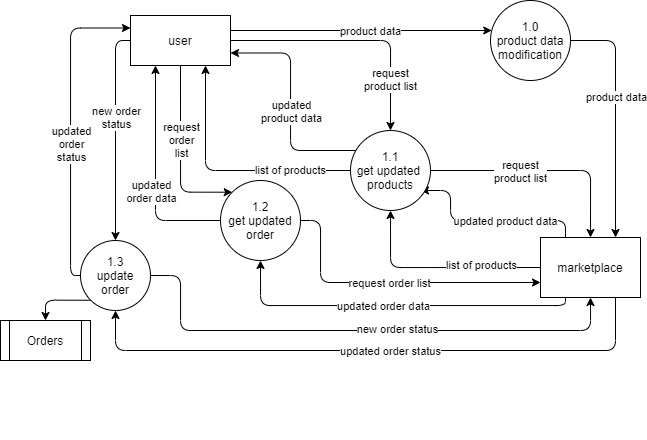


**Gambar 3. 2**

**Gambar Context Diagram Sistem**

Context diagram adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Pada Gambar 3.2 dapat dilihat terdapat lingkaran di tengah yang merupakan sebuah proses dari website integrasi marketplace iFurnHolic. Proses di tengah tersebut mendapatkan input dan output dari 2 (dua) pengguna yaitu pengguna dan marketplace. Panah yang mengarah masuk merupakan panah yang menunjukkan input, dan yang mengarah keluar merupakan panah yang menunjukkan output. Dengan demikian akan terlihat dengan jelas perpindahan data yang terjadi pada sistem.

Proses pada awalnya diawali dengan melakukan tambah, ubah, atau hapus produk. Pada proses tersebut, data-data yang dimasukkan akan langsung diarahkan ke marketplace menggunakan API. Pengguna akan mendapatkan kembali daftar produk secara lengkap yang dapat dilihat pada halaman utama. Segala perubahan yang terjadi akan terlihat dalam halaman tersebut. Pihak pengguna juga akan mendapatkan data pesanan dari marketplace yang diambil menggunakan API. Dengan demikian pengguna akan dapat melakukan pengolahan terhadap data pesanan yang didapatkan. Website ini ditujukan untuk digunakan oleh admin perusahaan, maka dari itu admin dapat menggunakan semua fitur yang tersedia serta tidak dibutuhkan pembedaan *role*. Fitur-fitur utama dari website ini adalah mengolah data produk serta mendapatkan data pesanan dari marketplace.



**Gambar 3. 3**

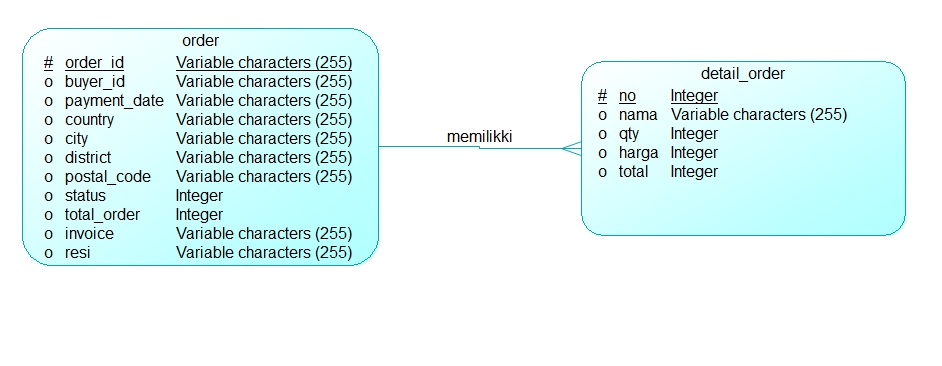
**DFD Level 0 Sistem**

Gambar 3.3 merupakan gambar diagram yang lebih detail mengenai sistem website integrasi marketplace iFurnHolic. Pengguna dapat melakukan perubahan data yaitu menambahkan, mengubah, atau menghapus data. Pengguna juga dapat melihat daftar produk yang ada. Selain itu pengguna dapat mendapatkan data pesanan yang ada. Dari daftar pesanan yang ada, pengguna dapat mengambil pesanan dan melakukan perubahan status pesanan apabila pesanan sudah atau sedang diproses.

1. **Desain Database**

Desain *database* adalah proses menghasilkan perincian model data dari suatu database tertentu. Model data tersebut dapat digunakan sebagai pengarah dalam membuat database untuk suatu program yang akan dibuat. Suatu model data bisa terdiri dari beberapa tabel database yang dapat dihubungkan satu dengan lainnya. Pembuatan desain database pada website integrasi marketplace iFurnHolic menggunakan Entity Relational Diagram atau yang dikenal dengan ERD.

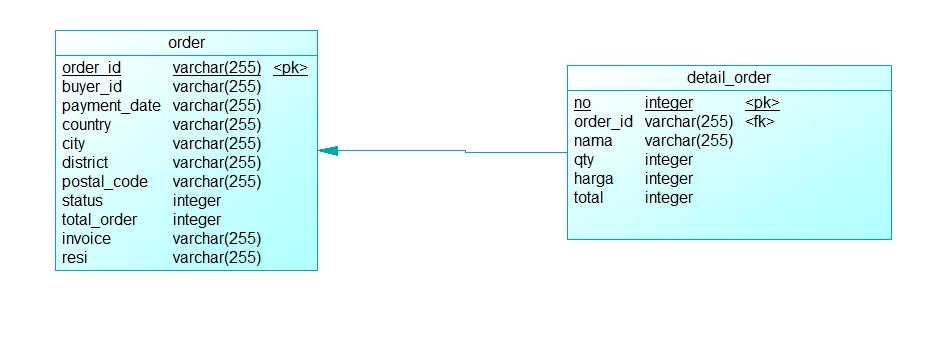
ERD adalah sebuah model untuk menyusun database agar dapat menggambarkan data yang mempunyai relasi dengan database yang akan didesain. Entitas adalah suatu objek yang dapat didefinisikan dalam lingkungan pemakai, yang penting bagi pemakai dalam konteks sistem yang akan dibuat, dan dapat didefinisikan secara unik atau saling berbeda. Atribut berfungsi untuk mendeskripsikan karakteristik dari entitas tersebut. Relasi adalah hubungan antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Garis merupakan penghubung antar atribut untuk menunjukkan hubungan entitas pada diagram ERD.



**Gambar 3. 4**

**Conceptual Data Model Sistem**

Conceptual Data Model (CDM) yang ditunjukkan pada Gambar 3.4 merupakan gambar desain database yang digunakan pada website integrasi marketplace iFurnHolic. Pada website ini hanya dibutuhkan 2 (dua) buah tabel untuk penyimpanan data pemesanan beserta detail dari pemesanan tersebut. Relasi antara kedua tabel adalah *one to many* atau yang sering disingkat dengan 1:N yang berarti sebuah order dapat memiliki lebih dari satu detail order. Hal ini dikarenakan dalam sebuah pembelian, seorang konsumen dapat membeli lebih dari 1 (satu) jenis barang. Tabel order terhubung dengan detail order melalui order\_id yang bersifat sebagai *foreign key*. Setiap tabel yang digunakan diberikan id masing-masing untuk memudahkan penyimpanan serta pengambilan data yang diinginkan.



**Gambar 3. 5**

**Physical Data Model Sistem**

Gambar 3.5 merupakan gambar Physical Data Model atau yang sering disingkat dengan PDM. PDM merupakan sebuah model skema yang mengimplementasikan CDM atau konsep pertama sebuah database agar siap diimplementasikan menjadi database yang sebenarnya. Hasil dari PDM-lah yang biasanya digunakan pada database yang sesungguhnya.

1. **Desain Tabel**

Desain tabel adalah suatu desain yang digunakan sebagai referensi dalam pembuatan database dari suatu program atau aplikasi. Tujuan dari pembuatan desain tabel adalah menunjukkan kepada pembuat aplikasi atribut apa saja yang dimiliki oleh suatu tabel dalam database yang ingin digunakan. Desain tabel dapat dibuat dengan Entity Relationship Diagram untuk dapat menunjukkan hubungan dari suatu tabel dengan tabel lainnya di dalam sebuah database.

Pada website integrasi marketplace iFurnHolic, tabel database tidak digunakan dalam semua modul, melainkan hanya pada fitur pengelolaan pemesanan. Tabel-tabel berikut hanya akan menyimpan data pemesanan. Fitur-fitur lain yang terdapat pada website tidak akan menggunakan database. Berikut merupakan tabel-tabel yang digunakan dalam website integrasi marketplace iFurnHolic.

**Tabel 3. 1**

**Struktur Tabel Order**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Tipe | Ukuran | Keterangan |
| order\_id | Varchar | 255 | Primary Key, Tidak boleh kosong |
| buyer\_id | Varchar | 255 | Tidak boleh kosong |
| payment\_date | Varchar | 255 | Tidak boleh kosong |
| country | Varchar | 255 | Tidak boleh kosong |
| city | Varchar | 255 | Tidak boleh kosong |
| district | Varchar | 255 | Tidak boleh kosong |
| postal\_code | Varchar | 255 | Tidak boleh kosong |
| status | Integer | 11 | Tidak boleh kosong |
| total\_order | Integer | 11 | Tidak boleh kosong |
| invoice | Varchar | 255 | - |
| resi | Varchar | 255 | - |

Tabel 3.1 merupakan struktur tabel penyimpanan pemesanan atau order. Data yang perlu dimasukkan dapat dibagi menjadi beberapa bagian yaitu detail konsumen, rincian pembayaran, dan tentu saja nomor invoice. Detail konsumen akan berisi id dari buyer tersebut, termasuk provinsi, kota/ kabupaten, kecamatan, kelurahan dan kode pos yang merupakan alamat dari pemesan. Bagian berikutnya yaitu rincian pembayaran, meliputi total transfer serta tanggal pemesanan. Selain itu, nomor resi dari pesanan tersebut juga akan dicatat apabila pengiriman telah dilakukan. Nomor resi yang dicatat kemudian akan diinformasikan ke pembeli juga. Tabel ini akan digunakan ketika ingin melakukan proses penyimpanan pesanan.

**Tabel 3. 2**

**Struktur Tabel Detail Order**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Tipe | Ukuran | Keterangan |
| no | Integer | 11 | Auto increment, tidak boleh kosong |
| order\_id | Varchar | 255 | Foreign key, tidak boleh kosong |
| nama | Varchar | 200 | Tidak boleh kosong |
| qty | Integer | 11 | Tidak boleh kosong |

**Tabel 3.2**

**(Lanjutan)**

|  |  |  |  |
| --- | --- | --- | --- |
| Field | Tipe | Ukuran | Keterangan |
| harga | Integer | 11 | Tidak boleh kosong |
| total | Integer | 11 | Tidak boleh kosong |

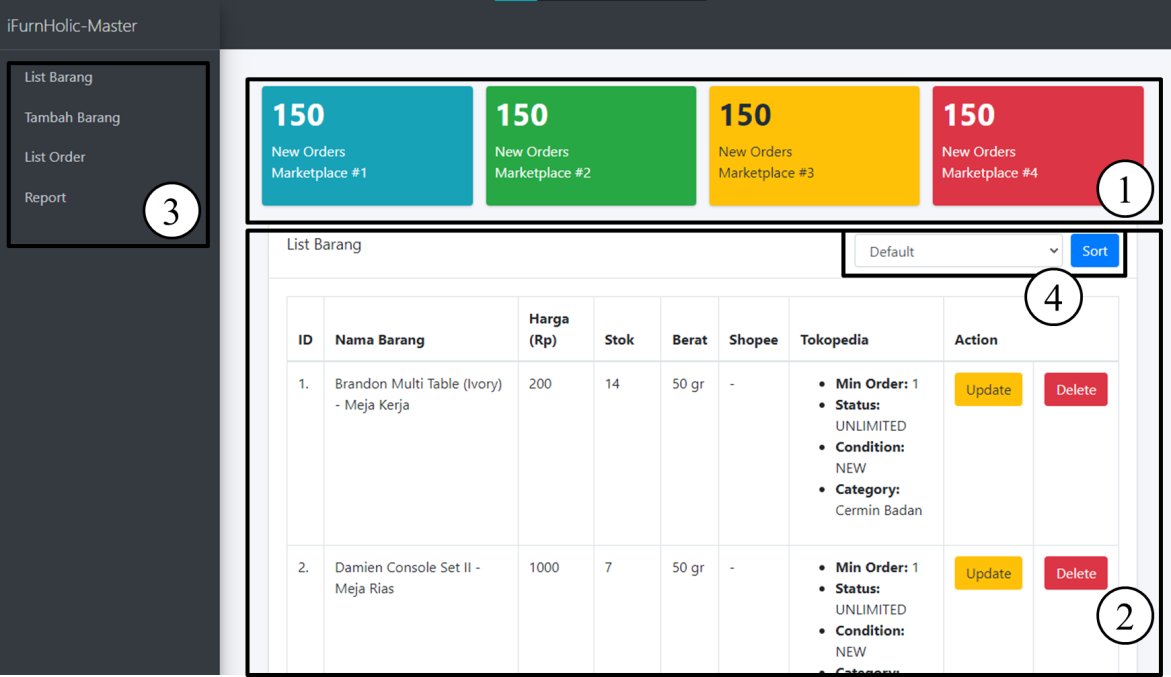
Tabel 3.2 merupakan struktur tabel penyimpanan detail order dimana tabel tersebut terhubung dengan tabel penyimpanan order dan dihubungkan dengan order\_id yang bersifat foreign key. Tabel ini akan digunakan sebagai pelengkap dari Tabel 3.1. Tabel ini menyimpan data berupa nama barang, banyak barang yang dibeli, harga satuan dari barang tesebut, dan subtotal dari pembelian barang yang bersangkutan. Dengan adanya tabel detail, maka suatu pemesanan dapat mencakup lebih dari 1 (satu) barang dan akan tercatat lebih jelas untuk subtotal dari masing-masing barang.

1. **Desain Interface**

Desain interface adalah pembuatan tampilan dari suatu program atau aplikasi yang akan dibuat. Desain interface dibuat agar pengguna dapat menggunakan suatu program atau aplikasi. Pembuatan tampilan website integrasi marketplace iFurnHolic menggunakan komponen-komponen dari AdminLte. Pada subbab ini akan dijelaskan dan dirincikan tampilan dari semua halaman pada website integrasi marketplace iFurnHolic.

1. **Tampilan Halaman Awal**

Tampilan halaman awal adalah tampilan yang akan selalu muncul saat pengguna membuka website integrasi marketplace iFurnHolic. Pada website ini, tampilan awalnya berupa jumlah total pesanan untuk masing-masing marketplace, serta daftar barang beserta detail barang untuk masing-masing marketplace. Namun karena untuk saat ini hanya terhubung dengan sebuah marketplace, maka data yang ditampilkan pada halaman ini masih belum terhubung dengan marketplace lain. Daftar barang akan ditampilkan dalam bentuk tabel. Pada website ini tidak terdapat halaman login ataupun register karena hanya digunakan oleh admin itu sendiri.

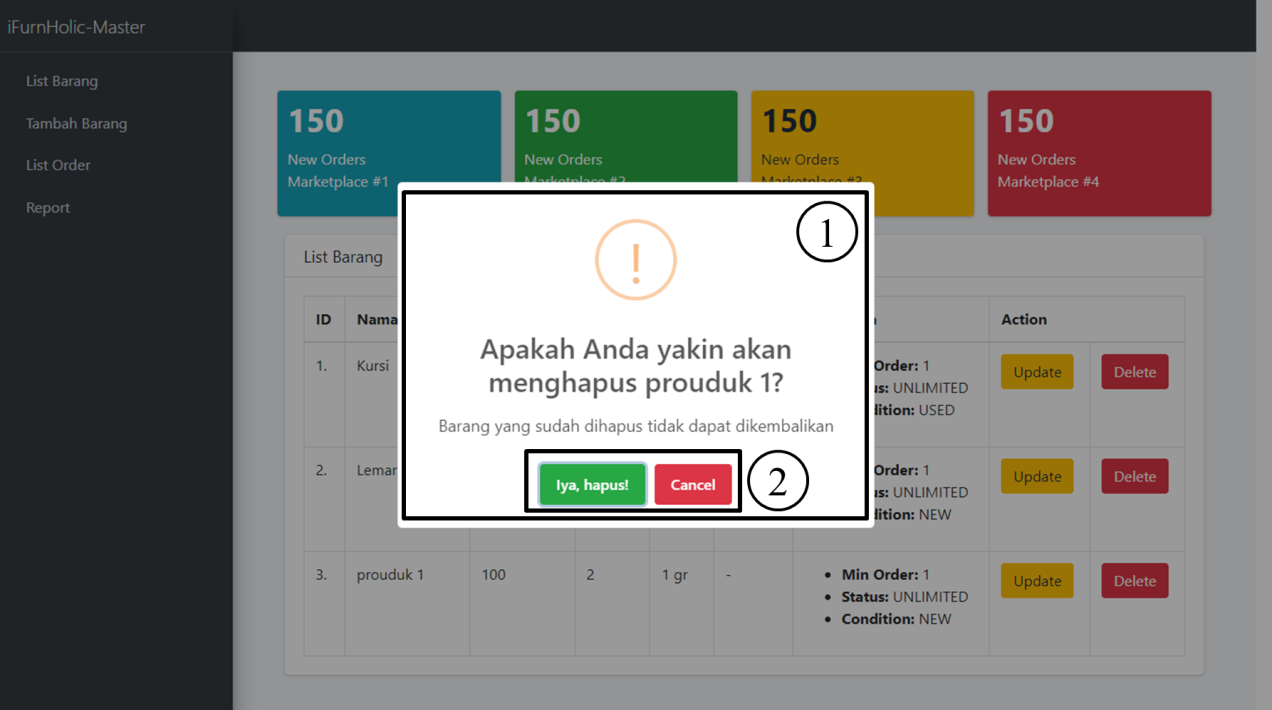
****

**Gambar 3. 6**

**Tampilan Halaman Awal Sistem**

Pada Gambar 3.6 dapat dilihat bahwa terdapat 4 (empat) bagian utama yaitu navigation bar, jumlah total pesanan, serta tabel data barang. Jumlah total pesanan dapat dilihat pada bagian yang ditandai dengan nomor 1 (satu). Jumlah total pesanan dibedakan untuk masing-masing marketplace. Namun sistem ini masih dalam pengembangan sehingga tidak semua marketplace terhubung, Tabel barang dapat dilihat pada bagian yang ditandai dengan nomor 2 (dua). Data barang akan memiliki detail berupa nama barang, harga, jumlah stok, serta berat barang dalam kilogram atau gram. Masing-masing marketplace akan memiliki detailnya masing-masing. Detail-detail tambahan dari masing-masing marketplace akan ditampilkan dalam bentuk list. Apabila tombol edit ditekan, maka data barang akan diarahkan ke halaman edit yang akan dijelaskan lebih lanjut pada tampilan pengelolaan barang. Pengguna juga dapat menghapus barang pada tabel dengan menekan tombol delete. Pengguna akan mendapatkan konfirmasi ketika sebuah barang dihapus. Konfirmasi berupa pesan *alert* dengan tombol konfirmasi.

Pengguna dapat berpindah-pindah halaman dengan menekan pilihan menu pada bagian kiri halaman. Fitur ini ditunjukkan dengan nomor 3 (tiga) pada Gambar 3.6. Terdapat 4 (empat) pilihan halaman yaitu list barang atau tampilan awal, tambah barang, list order, serta halaman laporan. Pengguna juga dapat mengurutkan data barang berdasarkan nama (A-Z dan Z-A), serta harga dari tertinggi atau terendah. Fitur ini ditunjukkan dengan nomor 4 (empat).



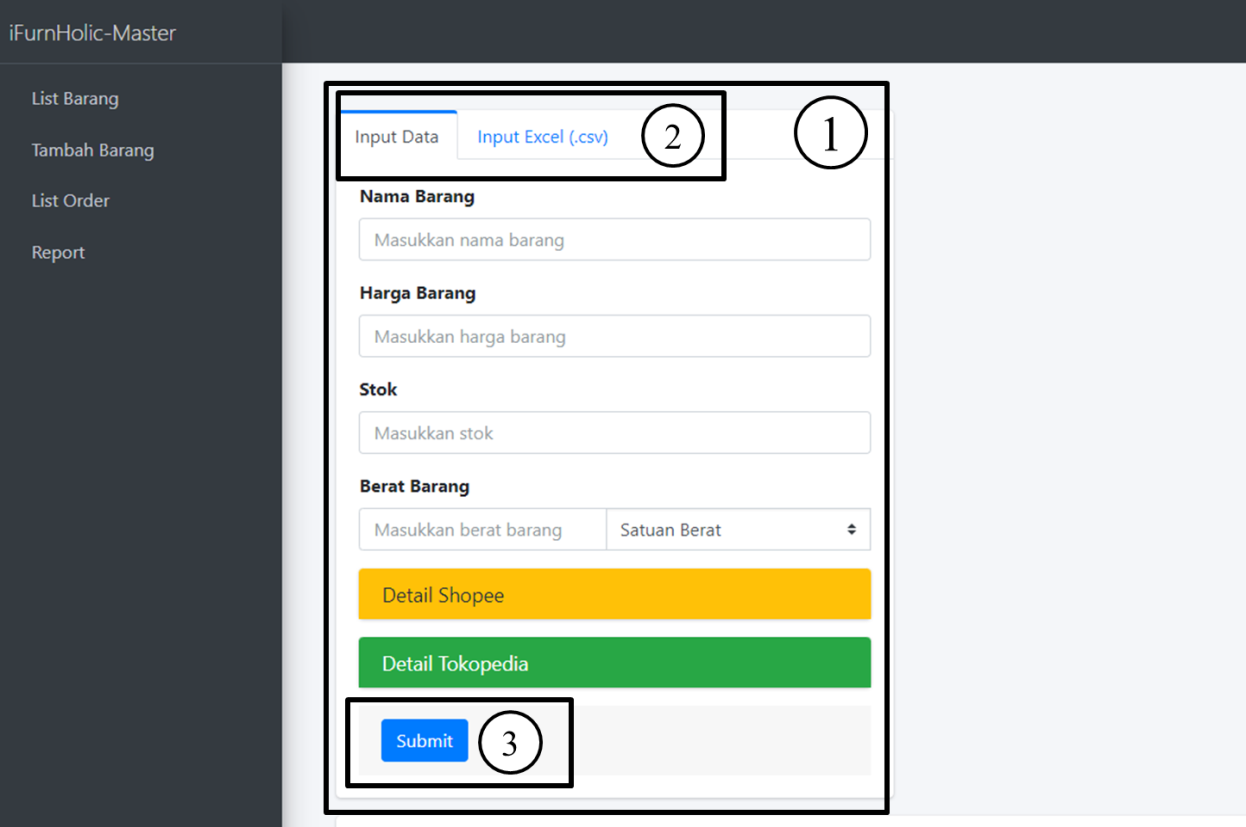
**Gambar 3. 7**

**Tampilan Hapus Barang**

Gambar 3.7 merupakan tampilan apabila tombol hapus pada salah satu barang ditekan. Website akan memunculkan sebuah peringatan seperti yang ditunjukkan pada nomor 1 (satu) pada Gambar 3.7. Apabila pengguna yakin ingin menghapus barang tersebut, maka pengguna dapat menekan tombol “Iya, hapus!” atau tombol yang berwarna hijau. Apabila produk sedang dalam transaksi aktif, maka meskipun pengguna menekan tombol hijau, produk tidak akan terhapus dan akan terdapat notifikasi. Apabila pengguna tidak ingin menghapusnya secara permanen, maka pengguna dapat menekan tombol cancel yang berwarna merah. Tampilan tombol tersebut ditunjukkan pada nomor 2 (dua) pada Gambar 3.7. Pengguna harus berhati-hati saat menghapus barang karena barang yang dihapus pada website akan secara otomatis terhapus pada marketplace dan tidak dapat dikembalikan lagi.

1. **Tampilan Fitur Pengelolaan Barang**

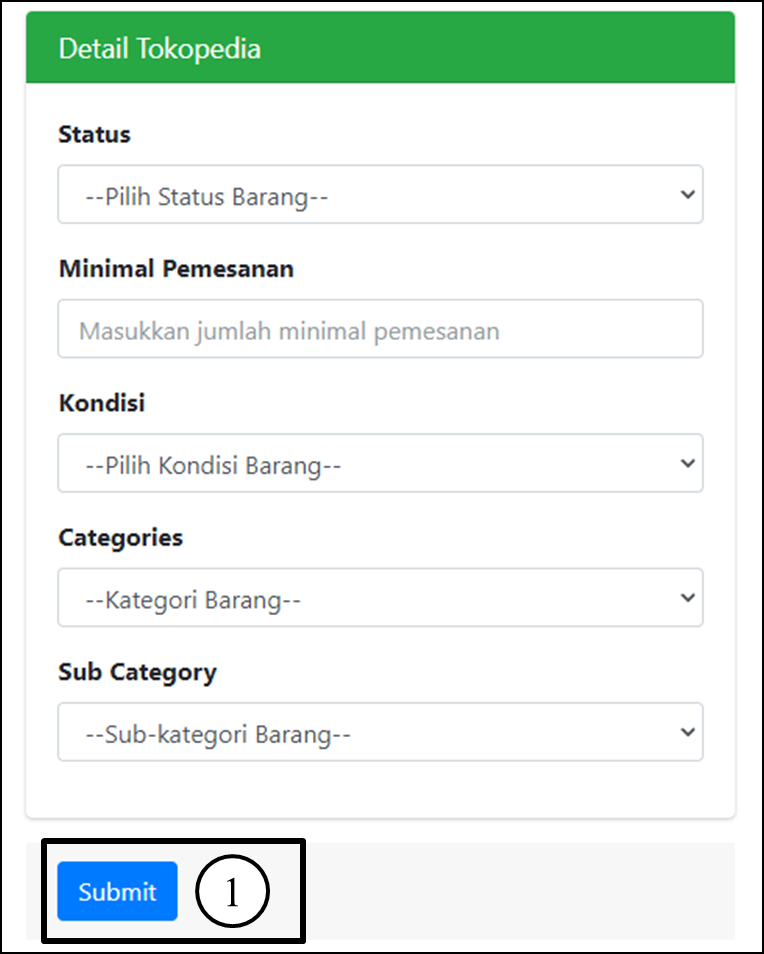
Fitur pengelolaan barang merupakan fitur dimana pengguna dapat melakukan perubahan terhadap data barang. Perubahan yang dapat dilakukan antara lain adalah menambahkan, atau merubah detail barang. Semua penambahan dan perubahan akan ditampung terlebih dahulu sebelum dirubah secara permanen pada marketplace yang dituju.



**Gambar 3. 8**

**Tampilan Halaman Pengelolaan Barang**

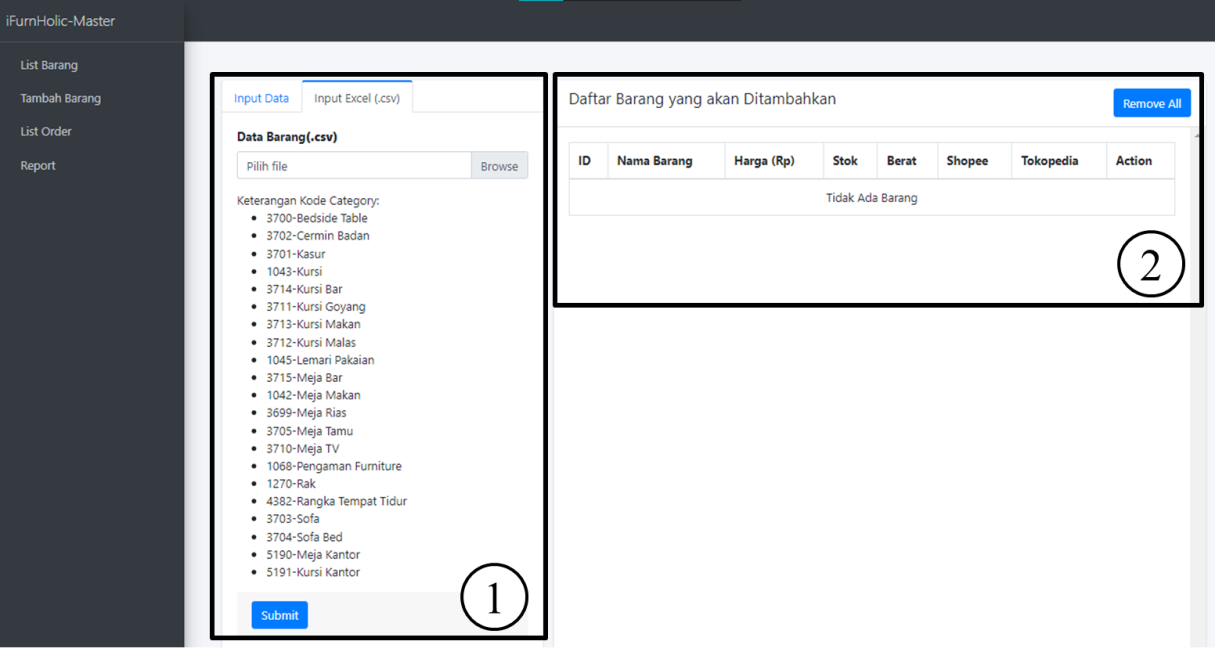
Pengguna dapat mengisikan data barang yang ingin ditambahkan secara manual pada bagian nomor 1 (satu) dari Gambar 3.8. Jika tidak ingin memasukkan secara manual, pengguna dapat memilih untuk memasukkan dengan menggunakan file bertipe .csv. Untuk menggunakan fitur tersebut, pengguna dapat memilih pada bagian nomor 2 (dua) pada Gambar 3.8. Data barang yang perlu dimasukkan secara umum antara lain adalah nama barang, harga, stok, serta berat barang dan satuan berat yaitu gram (gr) atau kilogram (Kg). Semua data wajib diisikan. Apabila sudah diisikan, maka pengguna dapat menekan tombol submit yang ditunjukkan dengan nomor 3 (tiga).



**Gambar 3. 9**

**Tampilan Detail Marketplace**

Masing-masing marketplace tentu memiliki ketentuan detail barang tersendiri. Maka dari itu tampilan untuk menambahkan detail dari tiap-tiap marketplace dibedakan. Gambar 3.9 merupakan contoh tampilan detail untuk Tokopedia. Pengguna dapat menambahkan detail-detail yang dibutuhkan dengan mengisi kolom-kolom yang ada. Sebagai contohnya untuk marketplace Tokopedia, terdapat tambahan data seperti status barang, jumlah minimal pemesanan, kondisi barang (baru atau sudah terpakai), kategori barang, serta sub-kategori barang. Setelah selesai mengisi, pengguna dapat menekan tombol submit yang ditunjukkan dengan nomor 1 (satu). Data yang berhasil melewati pengecekkan akan ditambahkan kedalam sebuah tabel sementara agar pengguna dapat melakukan perubahan lagi jika diinginkan.

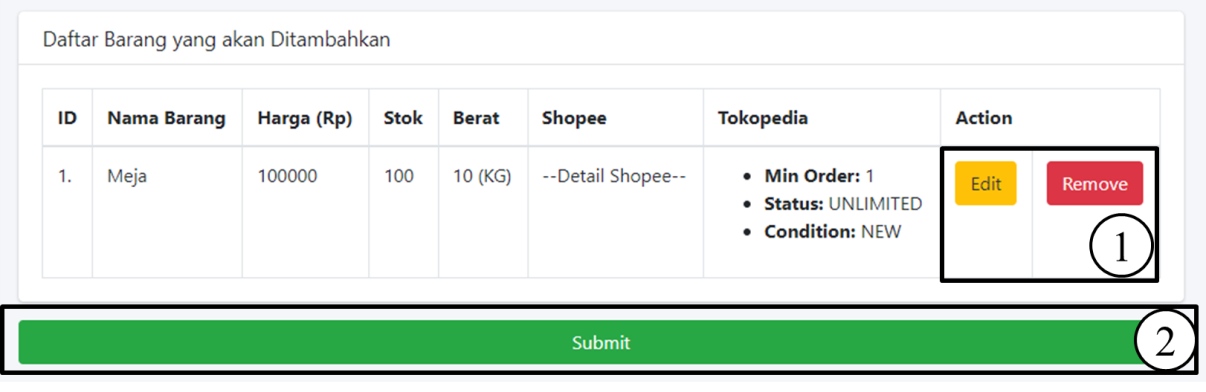


**Gambar 3. 10**

**Tampilan Input CSV**

Apabila tidak ingin melakukan penambahan barang secara manual dengan mengisi kolom satu per satu, pengguna dapat menambahkan barang secara masal dengan mengunggahnya pada bagian nomor 1 (satu). Format penulisan dalam file .csv akan ditentukan dari awal agar pengolahan datanya tidak membingungkan. Pada website juga telah disediakan kode untuk kategori yang akan digunakan. Setelah mengunggahnya, pengguna dapat menekan tombol submit berwarna biru.

Pengguna hanya dapat menambahkan barang secara masal ke dalam 1 (satu) marketplace. Apabila pengguna ingin menambahkan barang tersebut ke marketplace lainnya, maka pengguna harus melakukan update pada barang yang diinginkan untuk mengisi data tambahan seperti yang telah dijelaskan sebelumnya. Data barang yang berhasil diunggah akan masuk ke dalam tabel sementara yang ditunjukkan pada nomor 2 (dua). Format yang digunakan pada tabel sementara sama dengan tampilan tabel pada halaman utama agar tidak membingungkan pengguna. Data yang terdapat pada tabel sementara belum dimasukkan ke dalam marketplace sehingga pengguna masih dapat melakukan perubahan-perubahan.



**Gambar 3. 11**

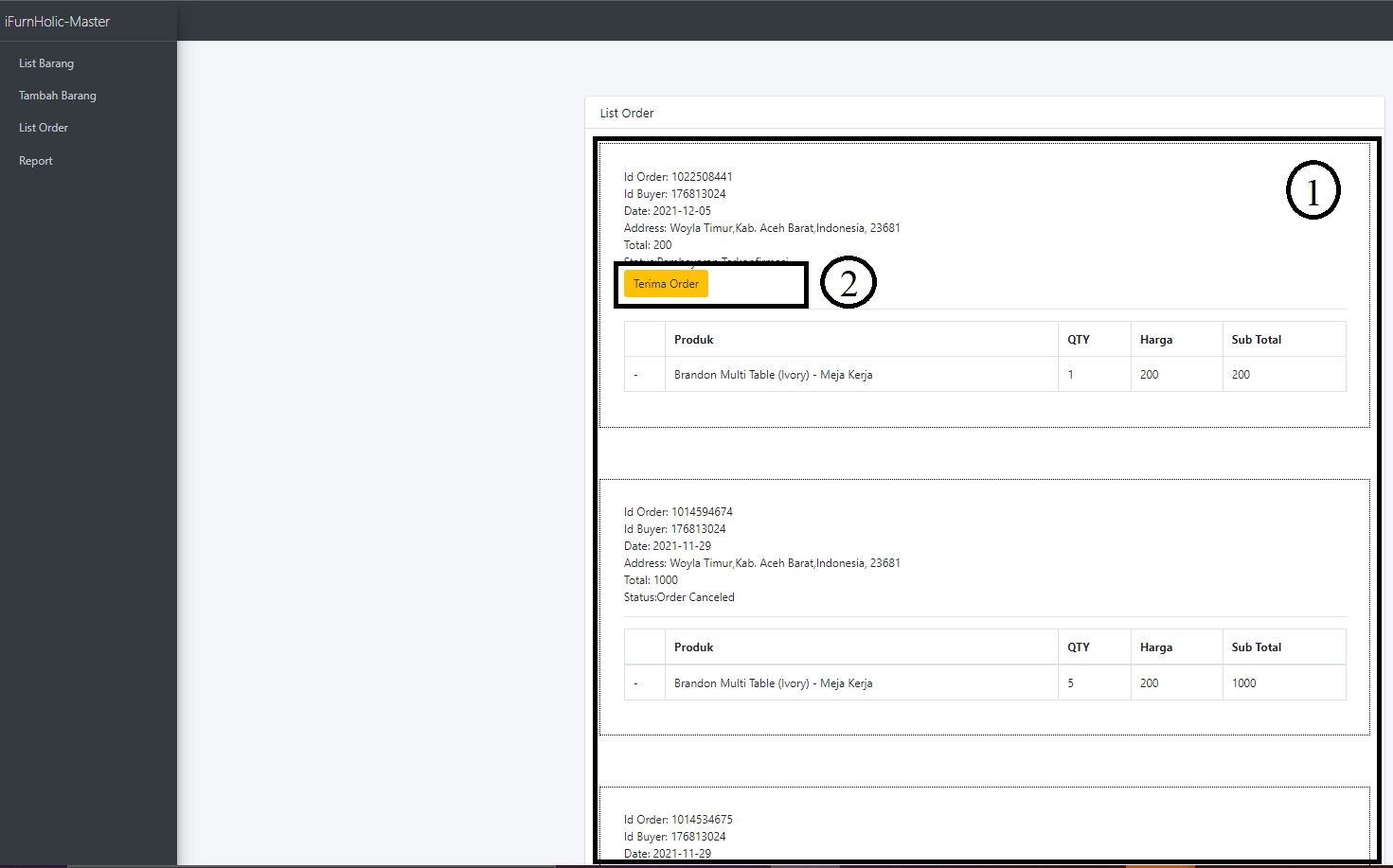
**Tampilan Data Barang yang akan Dimasukkan**

Gambar 3.11 merupakan tabel yang berisi data barang yang akan ditambahkan ke dalam marketplace. Sebelum mengunggahnya ke dalam marketplace, pengguna dapat melakukan perubahan terlebih dahulu dengan menekan tombol edit atau remove yang ditunjukkan dengan nomor 1 (satu). Apabila pengguna menekan tombol edit, maka data akan diarahkan kembali ke dalam form dan pengguna dapat melakukan perubaha. Apabila pengguna menekan tombol remove, maka barang akan dihapus dari tabel sementara.

Ketika pengguna merasa data yang akan ditambahkan sudah sesuai, pengguna dapat menekan tombol submit yang ditunjukkan dengan nomor 2 (dua). Setelah tombol submit ditekan maka pengguna akan diarahkan kembali ke halaman awal. Barang-barang yang berhasil ditambahkan akan muncul pada tabel di halaman awal dan secara otomatis ditambahkan ke dalam marketplace yang diinginkan.

1. **Tampilan Fitur Pengelolaan Pesanan**

Fitur pengelolaan pesanan merupakan fitur yang menyimpan semua data pesanan yang didapatkan oleh perusahaan dari semua marketplace yang terhubung dengan website. Pengguna dapat melihat semua pesanan yang belum diproses pada halaman ini. Pada halaman ini, pengguna juga dapat melakukan pemrosesan pesanan yang didapatkan. Data yang didapatkan akan ditampilkan dalam bentuk tabel dilengkapi dengan detail-detail pesanan.



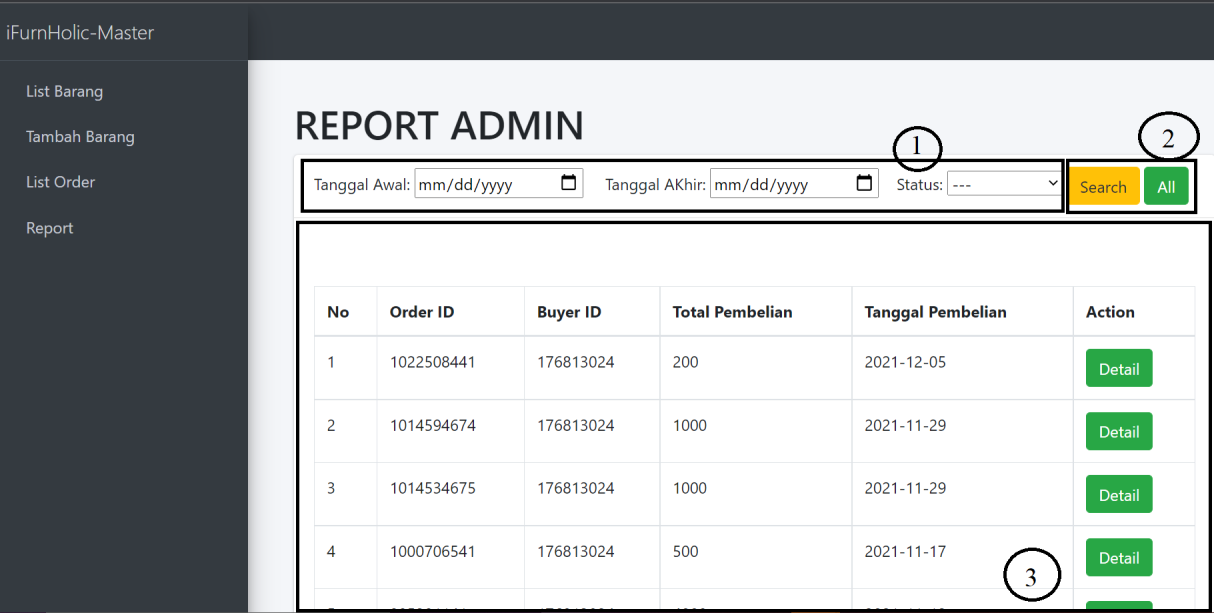
**Gambar 3. 12**

**Tampilan Halaman Pengelolaan Pesanan**

Gambar 3.12 merupakan gambar tampilan halaman pengelolaan pesanan. Semua pesanan yang didapatkan dari semua marketplace yang terhubung dengan website akan ditampilkan pada tabel seperti yang dapat dilihat pada nomor 1 (satu). Data yang dapat dilihat adalah nama barang, jumlah barang, id pembeli, serta status pesanan tersebut. Pengguna dapat memproses pesanan lebih lanjut dengan menekan tombol yang ditunjukkan dengan nomor 2 (dua). Dengan menekan tombol “Take Order” yang ditunjukkan dengan nomor 2 (dua), maka pengguna melakukan perubahan status pada marketplace tersebut dan dianggap telah melakukan pemrosesan pesanan.

1. **Tampilan Halaman Laporan**

Halaman laporan merupakan halaman dimana pengguna dapat melihat hasil dari penjualannya. Laporan pada website integrasi marketplace iFurnHolic akan disajikan dalam bentuk tabel. Pengguna dapat melakukan filtering pada tabel sesuai dengan jangka waktu yang diinginkan, atau sesuai dengan marketplace yang diinginkan.



**Gambar 3. 13**

**Gambar Tampilan Halaman Laporan**

Gambar 3.13 merupakan gambar tampilan halaman laporan website. Data pesanan yang sudah selesai akan ditampilkan pada tabel seperti yang ditunjukkan dengan nomor 3 (tiga). Pengguna dapat melihat id customer, id dari pesanan tersebut, total pembelian, serta tanggal pembelian. Pengguna juga dapat memilih jangka waktu dengan memilih tanggal awal dan akhir serta memilih status dari pesanan. Fitur ini ditunjukkan dengan nomor 1 (satu) pada Gambar 3.13. Secara *default*, laporan akan menampilkan semua data tanpa filter. Setelah memilih filter, maka pengguna dapat menekan tombol search yang ditandai dengan nomor 2 (dua). Jika pengguna ingin menghilangkan filter, maka pengguna dapat menekan tombol all pada nomor 2(dua).