

## Analisis Contingency Plan Sistem Billing Impor Menggunakan Metode User Centered Design (Studi Kasus : Terminal Petikemas Jakarta)

Ilham Dwi Prasetyo Nugroho, Rd. Nuraini Siti Fathonah

Program Studi D4 Teknik Informatika

Politeknik Pos Indonesia, Jalan Sariasih No. 54, Bandung, Indonesia

Email\_nuraini@poltekpos.ac.id

### ABSTRAK

Sebuah sistem pasti memiliki sebuah kelemahan dalam performa. Baik dari jaringan ataupun dari *hardware* maupun *software*. Perusahaan harus mencermati dengan seksama atas kelemahan sistem tersebut. Oleh karena itu, antisipasi yang perusahaan lakukan adalah membuat *contingency plan* dari sistem tersebut. Terutama perusahaan dibidang jasa pengiriman, transaksi harus dilakukan 24 jam nonstop. Sebelum *contingency plan* tersebut berjalan, perusahaan harus mengetahui seberapa sering sistem tersebut gangguan, dan mengetahui dimana letak gangguan yang sering terjadi. Oleh sebab itu, dibutuhkan sebuah kuesioner yang dilakukan perusahaan terhadap para pengguna. E-Kuesioner merupakan solusi untuk mempermudah perusahaan dalam menerima data yang dibutuhkan untuk terciptanya *contingency plan*.

**Kata kunci :** sistem, *Contingency plan*, E-Kuesioner,

### 1. PENDAHULUAN

Transaksi perdagangan luar negeri yang sering disebut ekspor impor hakikatnya merupakan transaksi sederhana dan tidak lebih dari jual atau beli barang perusahaan dengan perusahaan yang berada dinegara berbeda. Namun, pertukaran barang maupun jasa yang menyeberangi laut ataupun darat itu tidak jarang timbul permasalahan yang kompleks antara pengusaha yang mempunyai kebudayaan, bahasa dan adat yang berbeda-beda (Roselyne Hutabarat, 1997).

Ekspor dan impor di suatu negara merupakan kegiatan dari perdagangan internasional yang sering terjadi untuk memenuhi kebutuhan yang tidak bisa di produksi di negara tersebut. Proses impor dan ekspor bisa berjalan karena ada penghubung antara importir maupun ekportir dengan pihak jasa pengiriman seperti pelabuhan. Menurut Ahmad Ahyar dalam situs <https://arahkata.pikiran-rakyat.com/ekbis/pr-128995357/kemenperin-nilai-peran-pelabuhan-sangat-penting-dalam-perdagangan-internasional>, "Kementerian Perindustrian menilai peran pelabuhan sangat penting dalam perdagangan internasional, karena sebagai pintu gerbang utama dalam proses bongkar muat ekspor dan impor."

Kota Jakarta memiliki banyak Pelabuhan yang bisa digunakan untuk kegiatan impor maupun ekspor, dan merupakan tempat strategis serta merupakan jalur perdagangan yang masuk ke Indonesia. Salah satu pelabuhan yang terbesar dan terkenal di Jakarta adalah Pelabuhan Tanjung Priok. Daerah Tanjung Priok sendiri ada beberapa perusahaan yang bergerak dibidang jasa pengiriman ekspor maupun impor, diantaranya : PT. Jakarta International Container Terminal, KSO. Terminal Petikemas Koj, PT. Multi Terminal Indonesia, serta yang baru dibuat untuk menambah terminal di Jakarta yaitu PT. New Port Container Terminal 1. Terminal Petikemas X merupakan salah satu perusahaan jasa dibidang pelayanan ekspor dan impor yang terdapat di

Daerah Jakarta Utara. Terminal Petikemas X sendiri bukan merupakan perusahaan perseroan terbatas melainkan perusahaan dalam bentuk Kerja Sama Operasi. Perusahaan ini dibentuk untuk membantu lancarnya proses ekspor dan impor yang dikhawatirkan, terminal Y tidak bisa menampung lonjakan arus barang container ekspor dan impor yang akan terjadi. Terminal Petikemas X sendiri akan menciptakan *smart digital* terminal yang mengembangkan inovasi digital. Inovasi digital yang dimaksud adalah Pelayanan *Billing* yang semula masih dilakukan secara manual *Billing*, sekarang sudah menjadi pelayanan *Billing* mandiri melalui fasilitas *E-Billing*, pembayaran melalui media elektronik dengan memanfaatkan kode *billing* sebagai kode transaksi (Dewi, Pratiwi, et al, 2019), serta pembayaran yang menggunakan EDC dari beberapa bank yang sudah melakukan kerjasama *Host to Host* (H2H).

*E-Billing* diharapkan bisa mempercepat transaksi sehingga transaksi dapat berjalan tanpa ada antrian ketika melakukan pembayaran. Tetapi, Ada salah satu kekurangan di dalam *E-Billing* tersebut yaitu ketika ada masalah di server tempat *E-Billing* beroperasi, transaksi akan terhambat dan bisa menjadi penumpukan pelanggan dalam melakukan pembayaran karena menunggu server *E-Billing* tersebut kembali normal. Untuk mengantisipasi hal tersebut terjadi, pihak perusahaan menyediakan alternatif lainnya berupa *contingency plan*.

*Contingency plan* atau plan darurat merupakan langkah yang diambil ketika server di *e-billing* utama mengalami gangguan sehingga terjadinya penghambatan transaksi para importir, *Contingency plan* yang akan di bangun merupakan sebuah aplikasi *billing* offline di perusahaan Terminal Petikemas X, ketika *e-billing* sudah kembali normal, data data yang ada di aplikasi *billing* offline akan bisa diinputkan ke server utama *e-billing* ketika sudah kembali normal.

Sebelum terciptanya *contingency plan*, terlebih dahulu dilakukan kuisioner terhadap user yang akan menggunakan sistem dari *contingency plan* tersebut. Kuisioner berfungsi untuk mengetahui apakah sebuah *contingency plan* dibutuhkan di sebuah perusahaan atau tidak.

## 2. TINJAUAN PUSTAKA

### 2.1 Sistem

Sistem adalah sebuah tatanan (keterpaduan) yang terdiri atas sejumlah komponen fungsional (dengan satuan fungsi dan tugas khusus) yang saling berhubungan dan secara bersama-sama bertujuan untuk memenuhi suatu proses tertentu. [1]

#### 2.1.1 Karakteristik Sistem

Untuk penjelasan yang lebih detail, membangun serta mengembangkan sesuatu sistem yang benar, maka perlu ada yang membedakan antara unsur-unsur dari sistem yang terbentuk, seperti Components, Boundary, Environment, Input, Output, Interfaces, Proses serta Objectives dan Goal. [2]

##### 1. Komponen Sistem (Components)

Proses sistem yang mengirimkan input menjadi bentuk setengah jadi berupa output.

##### 2. Batasan Sistem (Boundary)

Menggambarakan suatu unsur yang termasuk dalam sistem dan yang bukan termasuk dalam sistem atau di luar sistem.

##### 3. Lingkungan (Environment)

Lingkungan merupakan unsur yang berada di luar sistem. Lingkungan juga menyediakan asumsi, kendala serta inputan terhadap suatu sistem.

##### 4. Masukan Sistem (Input)

Inputan merupakan sumber daya (data, energi, peralatan) yang dibutuhkan dan dimanipulasi oleh suatu sistem.

##### 5. Keluaran Sistem (Output)

Output merupakan sumber daya (informasi, laporan, dokumen, tampilan layar maupun barang jadi) yang telah berubah dari proses input di dalam suatu sistem.

##### 6. Penghubung (Interface)

Interface merupakan tempat dimana pengguna sistem melakukan interaksi terhadap sistem.

##### 7. Pengolah Sistem (Process)

Sistem mempunyai satu bagian yaitu untuk pengolahan data dari input yang akan dikeluarkan menjadi output.

##### 8. Sasaran Sistem (Objective)

Sistem pasti akan memiliki tujuan dari terbentuknya sistem tersebut. Dari tujuan tersebut, sangat menentukan masukan yang dibutuhkan oleh sistem dan akan mengeluarkan outputan yang sesuai dengan tujuan sistem tersebut.

### 2.2 Impor

Impor adalah proses pembelian barang atau jasa asing dari suatu negara ke negara lain. Tingkat impor dipengaruhi oleh hambatan peraturan perdagangan (Nopirin 2011). Pemerintah mengenakan tarif (pajak) pada produk impor, pajak itu biasanya dibayar

langsung oleh importir, yang kemudian akan membebaskan kepada konsumen berupa harga lebih tinggi dari produknya, ketika pemerintah asing menerapkan tarif, kemampuan perusahaan asing untuk bersaing di negara-negara itu dibatasi. Pemerintah juga dapat menerapkan kuota pada produk impor, yang membatasi jumlah produk yang dapat di impor [3]

### 2.3 Unified Modeling Language

*Unified Modeling Language* merupakan metode pemodelan visual yang bertujuan untuk merancang atau membuat software berorientasi objek. *UML* merupakan bahasa visual untuk pemodelan bahasa berorientasi objek, maka elemen dan diagram yang berbasis pada paradigma objek oriented.

*Unified Modeling Language* juga salah satu tool untuk merancang pengembangan software yang berbasis object oriented. *Unified Modeling Language* juga memberikan standar penulisan dalam sebuah sistem blue print, yang meliputi konsep bisnis proses, penulisan kelas-kelas dalam bahasa program yang spesifik, skema database, dan komponen-komponen yang diperlukan dalam sistem software. [4]

### 2.4 Contingency Plan

Contingency plan adalah suatu perencanaan untuk kondisi yang tidak terduga. Perusahaan harus bersiap-siap untuk menghadapi, mendeteksi, bereaksi serta pemulihan dari pasca kondisi tersebut. Kondisi tersebut mengancam keamanan asset sebuah perusahaan. Tujuan dari contingency plan adalah melakukan perbaikan menuju mode operasi normal dengan biaya minimum serta gangguan gangguan aktivitas normal bisnis setelah terjadi bencana yang tak terduga.

### 2.5 Metode User Centered Design

*User centered design* (UCD) atau disebut *human centered design*. *User centered design* merupakan sebuah metode yang digunakan untuk pengembangan sistem secara interaktif terhadap pengguna yang bertujuan untuk membuat sebuah perangkat lunak atau sistem. [5]

### 2.6 Sonarlint

SonarLint merupakan tools ekstensi IDE yang dapat membantu mendeteksi serta memperbaiki masalah kualitas saat menulis sebuah kode program. SonarLint dapat menemukan kekurangan pada kode program tersebut sehingga kode program tersebut diperbaiki. [6]

## 3. Metodologi

Metodologi dalam penelitian ini menggunakan studi literatur untuk melengkapi bahan dari sistem yang akan di bangun, dengan mengumpulkan dari sumber internet, jurnal yang berhubungan dengan sistem.

### 3.1 Studi Literatur

Studi literatur dilakukan dengan mengkaji Pustaka dari buku-buku dan mengunjungi situs-situs internet untuk mendukung penelitian. Studi literatur yang dilakukan ialah materi :

- a. Metode User Centered Design
- b. Usability Testing

## c. Kuesioner

**3.2 Pengumpulan data**

Peneliti menggunakan metode observasi lapangan untuk menentukan permasalahan yang di butuhkan oleh perusahaan.

**3.3 Usability Testing**

*Usability testing* diterapkan pada sistem e-kuesioner yang akan dibangun. Untuk para pengguna akan dibuatkan e-kuisiioner yang harus diisi supaya diketahui apakah *contingency plan* dibutuhkan dalam perusahaan atau tidak.

*Usability Testing* memiliki 10 pertanyaan yang hasilnya akan dikalkulasikan dengan menggunakan rumus pencarian rata-rata. Pertanyaan tersebut adalah

Table 1 10 Pertanyaan Usability Testing

No	Pertanyaan
1	Saya berpikir akan menggunakan sistem ini lagi
2	Saya merasa sistem Selalu terjadi gangguan
3	Saya merasa sistem ini mudah digunakan
4	Saya sistem sering tidak bisa digunakan
5	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya
6	Saya merasa jaringan internet perusahaan selalu bermasalah
7	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat
8	Saya merasa sistem selalu <i>not responding</i>
9	Saya merasa tidak ada hambatan dalam menggunakan sistem ini
10	Saya merasa bingung ketika sistem sedang terjadi gangguan

Setelah membuat kuisiioner dengan pertanyaan tersebut, *usability testing* mempunyai 5 pilihan jawaban, yaitu sangat tidak setuju, tidak setuju, ragu-ragu, setuju, sangat setuju. Masing-masing jawaban memiliki skor dari 1 sampai 5. Berikut merupakan pilihan jawaban dari *usability testing*.

Table 2 responden Usability Testing

Jawaban	skor
Sangat Tidak Setuju (STS)	1
Tidak Setuju	2
Ragu-ragu	3
Setuju	4

**3.3.1 Perumusan Usabilitu Testing**

$$\bar{X} = \frac{\sum X}{n}$$

Keterangan Rumus diatas

$\bar{X}$  = Rata Rata

$\sum X$  = Jumlah Sekor Pertanyaan

N = Jumlah Repsonden

**3.3.2 Kategori Hasil Penilaian**

Table 3 kategori hasil

50-100	Tidak dibutuhkan
0-49	Sangat Dibutuhkan

**3.4 Perancangan**

Peneliti menampilkan perancangan berupa arsitektur sistem, usecase diagram, class diagram, dan perancangan *database*.

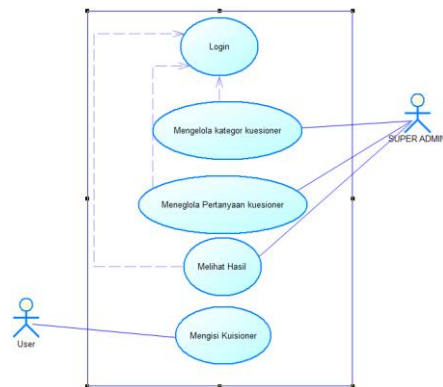
**3.4.1 Arsitektur Sistem**

Arsitektur sistem merupakan kebutuhan-kebutuhan berupa *hardware* maupun *software* dalam pembuatan sistem tersebut.

Pada penelitian ini, peneliti menggunakan spesifikasi *hardware* berupa laptop dengan *harddisk* 1 TB dengan *memory RAM* 8 GB serta *processor Intel® core™ i7-8750H CPU @2,20 Ghz*. Spesifikasi *software* menggunakan windows sebagai sistem operasi, *xampp* sebagai server basis data, VS.Code sebagai text editor untuk melakukan coding serta *Google Chrome* untuk browser.

**3.4.2 Usecase Diagram**

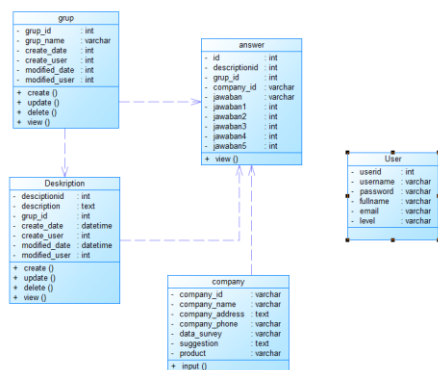
Berikut ini merupakan perancangan usecase yang terdapat di sistem yang akan dibangun.



Gambar 1. Usecase Diagram

### 3.3.3 Class Diagram

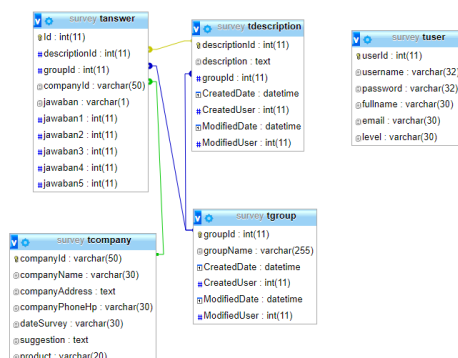
Class diagram merupakan gambaran hubungan kelas, atribut serta operasi yang terdapat di sistem. Perancangan class diagram dapat dilihat pada Gambar 2.



Gambar 2 Class Diagram

### 3.4.4 Perancangan Database

Perancangan database bertujuan memberi gambaran tentang tabel dan field yang di gunakan dalam sistem yang akan di bangun. Pada perancangan ini, menggambarkan tabel yang saling berelasi terhadap tabel yang lain.



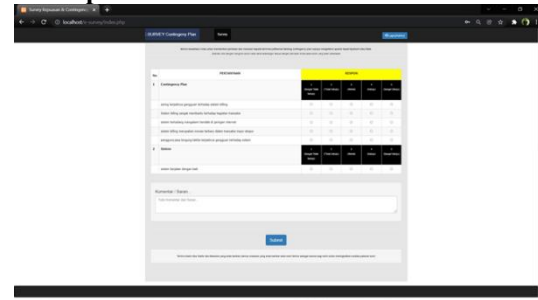
Gambar 3 Perancangan Database

## 4. Hasil dan Pembahasan

### 4.1 Analisis User Centered Design

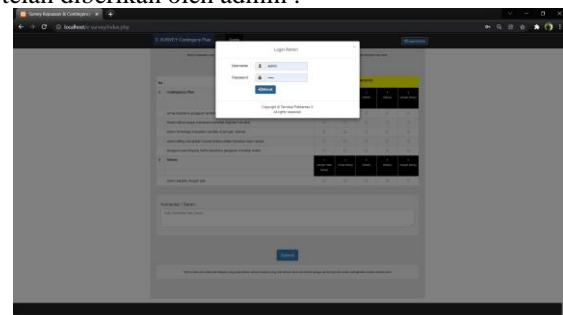
Peneliti menggunakan *usability testing* dalam metode *user centered design*. *Usability testing* diterapkan pada sistem e-kuesioner yang akan di bangun. Untuk para pengguna akan dibuatkan e-kuisiioner yang harus diisi supaya diketahui apakah contingency plan dibutuhkan dalam perusahaan atau tidak.

### 4.2 UI Aplikasi E-Kuesioner



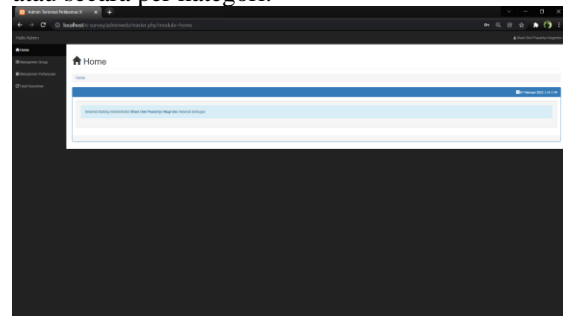
Gambar 4 halaman utama

Gambar ini merupakan tampilan *user interface* pengguna dalam mengisi kuesioner yang telah diberikan oleh admin .



Gambar 5 Halaman Login

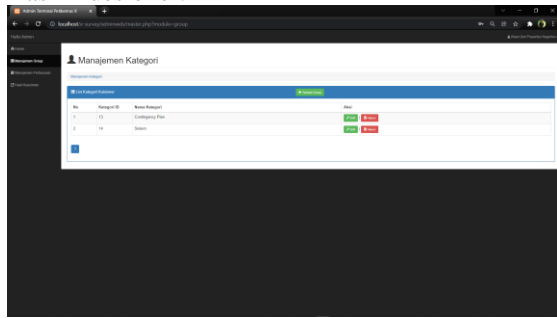
Gambar berikutnya ialah tampilan dari login admin. Halaman login diperuntukan untuk admin dari sistem ini. Admin akan melakukan login untuk dapat mengakses menu admin dan mengelola sistem ini. Admin dapat mengelola kategori pertanyaan, mengelola pertanyaan serta admin dapat mengetahui hasil dari kuesioner tersebut, baik secara keseluruhan atau secara per kategori.



Gambar 6 Menu Admin

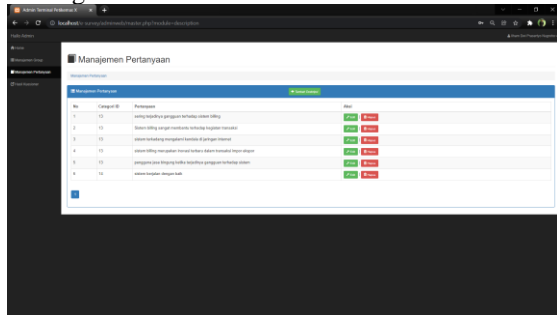
Gambar berikutnya ialah gambar dari menu admin. Dalam menu admin, ada berbagai macam yaitu

: 1. Manajemen Grup. 2. Manajemen Pertanyaan. 3. Hasil Kuesioner.



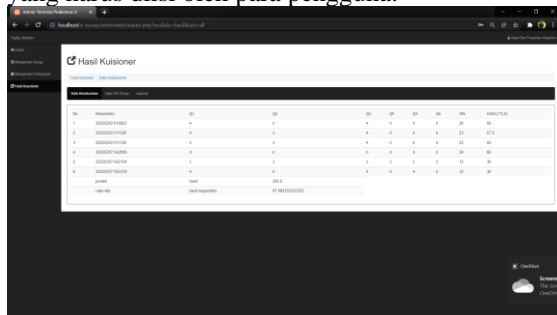
Gambar 7 Menu Kategori

Halaman manajemen kategori pertanyaan, admin bertugas untuk mengkategorikan pertanyaan tersebut. Tujuan mengkategorikan pertanyaan ialah, untuk memudahkan dalam melihat hasil dari pertanyaan yang sudah dikelompokkan atau dikategorikan



Gambar 8 Menu Manajemen Pertanyaan

Interface ini merupakan menu untuk manajemen pertanyaan atau mengelola pertanyaan. Dalam menu ini, admin bisa menambahkan pertanyaan, mengedit pertanyaan serta menghapus pertanyaan. Pertanyaan yang ada di menu ini, akan tampil di halaman utama pengguna sebagai kuesioner yang harus diisi oleh para pengguna.



Gambar 9 Menu Hasil Kuesioner

Pada menu ini, admin bisa melihat hasil dari kuesioner yang sudah dilakukan. Menu ini di bagi menjadi 3 bagian, yaitu : Data Keseluruhan, Data per kategori dan laporan. Bagian tersebut sama saja, yang membedakan adalah isi dari submenu tersebut. Untuk data keseluruhan adalah data yang di dapat dari kuesioner dari seluruh kategori. Jika hanya ada 1 kategori saja, maka bisa di pastikan hasilnya merupakan penentuan dibutuhkannya contingency

plan atau tidak. Jika admin membuat 2 kategori, makan bisa ke submenu yaitu data per kategori.

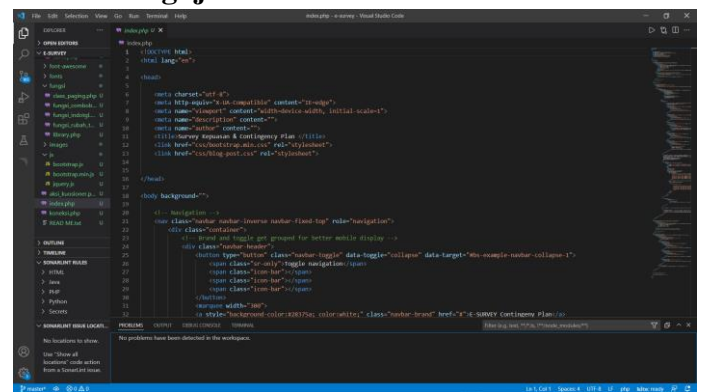
### 4.3 Pengujian

Pada tahap pengujian, peneliti melakukan 2 pengujian yaitu *white box* dan *usability testing*. Pengujian White box bertujuan untuk mengetahui apakah terjadi kesalahan terhadap codingan yang sudah di buat. Sedangkan untuk *usability testing* bertujuan menentukan penilaian untuk menentukan pentingnya contingency plan atau tidak

#### 4.3.1 White Box

Pada pengujian white box ini, peneliti menggunakan sonarlint untuk menguji codingan yang telah dibuat dalam membangun sistem e-kuesioner. Berikut beberapa pengujian yang dilakukan disetiap codingan yang telah dibuat :

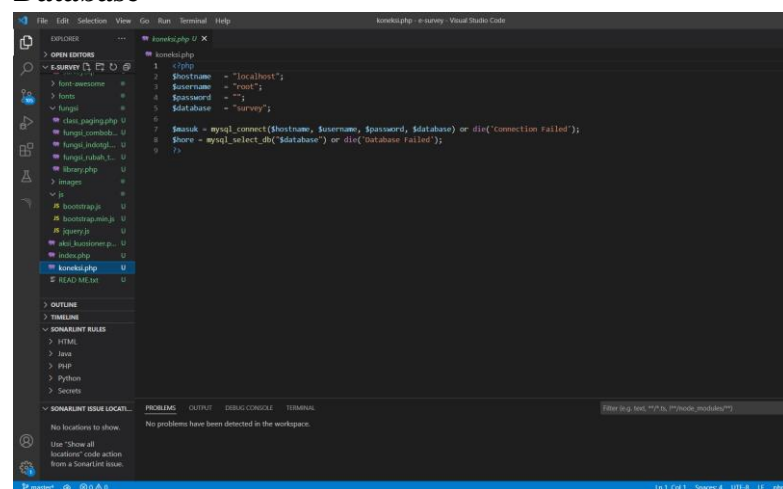
#### 4.3.1.1 Pengujian Halaman Utama



Gambar 10 contoh pengujian Sonarlint

Pada pengujian ini, sonarlint tidak mendeteksi adanya kesalahan yang terjadi pada codingan halaman utama yaitu tempat pengisian informasi pelanggan, tempat pengisian kuesioner dan saran serta tombol untuk melakukan login admin.

#### 4.3.1.2 Pengujian Terhadap Koneksi Database

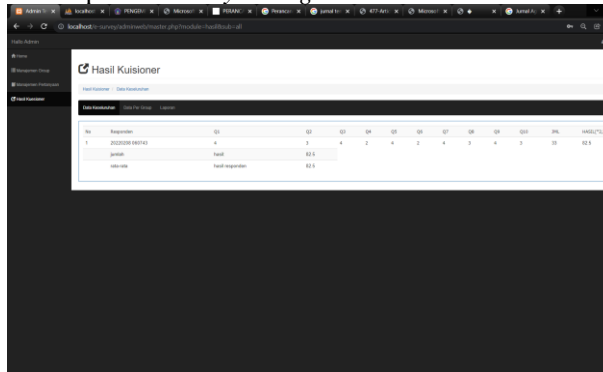


Gambar 11 Contoh Pengujian Sonarlint

Pada pengujian ini, codingan koneksi database tidak ditemukan kesalahan yang terjadi. Codingan ini merupakan pemanggilan data dari database yang akan ditampilkan dalam sistem ini.

### 4.3.2 Usability Testing

Pengujian *Usability testing* dilakukan pada sistem *e-kuesioner contingency plan*. Penerapan *usability testing* terdapat di menu hasil pada submenu data keseluruhan hingga data per grup. Ketika pengguna sudah menginputkan hasil dari kuesioner tersebut, maka secara otomatis sistem akan menghitung berdasarkan aturan aturan yang berlaku dalam *usability testing*. Berikut submenu yang menerapkan *usability testing* :



*Gambar 12 penerapan Usability Testing*

## 5. Kesimpulan dan Saran

Berdasarkan pembahasan ini, dapat disimpulkan bahwa menganalisis sebuah *contingency plan* sistem *billing* impor dibutuhkan sebuah kuesioner dari para pengguna. Kuesioner yang dipakai adalah kuesioner yang sudah terkomputerisasi sehingga tidak membuang-buang kertas. Dari sistem kuesioner

tersebut tidak hanya dapat digunakan ketika penentuan contingency plan saja, namun bisa digunakan untuk mengetahui sistem yang harus di perbaharui sehingga memudahkan pengguna untuk memakai sistem tersebut.

## DAFTAR PUSTAKA

- [1] Fathansyah, Basis Data, Bandung: Informatika Bandung, 2015.
- [2] H. A. Fatta, Analisis dan Perancangan Sistem Informasi untuk Keunggulan Bersaing Perusahaan dan Organisasi Modern, Yogyakarta: C.V. ANDI OFFSET, 2007.
- [3] R. B. Silitong, Z. Ishak and M. , "Pengaruh Ekspor, Impor, dan Inflasi Terhadap Nilai Tukar Rupiah di Indonesia," *Jurnal Ekonomi Pembangunan*, vol. 15, no. 1, pp. 53-59, 2017.
- [4] I. S. Y. Saputri, M. Fadhli and I. Surya, "Penerapan Metode UCD (User Centered Design) pada E-Commerce Putri Intan Shop Berbasis Web," *Jurnal Nasional Teknologi dan Sistem Informasi*, vol. 3, no. 2, pp. 269-278, 2017.
- [5] D. A. Wibowo, "Meningkatkan “Code Quality” dengan Plugin SonarLint di IntelliJ IDEA," *Javan Cipta Solusi*, 16 November 2020. [Online]. Available: <https://blog.javan.co.id/meningkatkan-code-quality-dengan-plugin-sonarlint-di-intellij-idea-36705b6cd8fa>. [Accessed 8 Februari 2022].