



APLIKASI PENGECEKAN DOKUMEN EKSPOR DI PT. XY BERBASIS VBA EXCEL MENGGUNAKAN METODE *ITERATIVE AND INCREMENTAL* DENGAN PENDEKATAN ARSITEKTUR MVC (*MODEL-VIEW-CONTROLLER*)

Firda Rosiana Tanjung^{1*}, Liptia Venica²

¹Program Studi Pendidikan Sistem & Teknologi Informasi, Universitas Pendidikan Indonesia, Indonesia

²Program Studi Mekatronika dan Kecerdasan Buatan, Universitas Pendidikan Indonesia, Indonesia

Abstrak: Dokumen ekspor menjadi salah satu bagian penting dalam keperluan proses ekspor barang. Hampir setiap perusahaan industri manufaktur yang aktif menjalankan operasi di sektor perdagangan internasional terlibat dalam kegiatan ekspor dengan alur yang kompleks termasuk dalam mengelola dokumen-dokumen ekspor tersebut. PT. XY merupakan salah satu perusahaan industri manufaktur di Indonesia yang memiliki Departemen Ekspor sebagai penanggung jawab dalam mengurus keperluan ekspor barang yang mereka produksi untuk dikirim ke luar negeri. Selain mengelola berbagai regulasi dan persyaratan ekspor barang, Departemen Ekspor PT.XY juga harus memastikan bahwa keperluan pendukung lainnya seperti dokumen-dokumen yang diperlukan telah lengkap. Untuk memenuhi hal tersebut, Pegawai Departemen Ekspor PT.XY melakukan pengecekan ketersediaan dokumen secara manual yang dirasa tidak efektif untuk dilakukan setiap hari karena meningkatnya volume dokumen yang akan terus bertambah setiap hari, akibatnya pengecekan memakan waktu yang cukup lama dan berkemungkinan besar dapat terjadi kesalahan pemeriksaan data. Dengan adanya permasalahan tersebut, maka Penulis melakukan analisis, membuat perancangan dan mengimplementasikannya menjadi sebuah Aplikasi Pengecekan Dokumen Ekspor berbasis VBA dan Macro Excel. Tujuan pembuatan aplikasi ini adalah untuk memudahkan Pegawai Departemen Ekspor PT. XY ketika memeriksa kelengkapan dokumen-dokumen ekspor tersebut. Aplikasi ini dapat diintegrasikan dengan *private FTP server* Departemen Ekspor sehingga menghasilkan sumber data yang valid dan relevan sesuai *current condition*. Aplikasi dibuat menggunakan metode pengembangan sistem *Iterative and Incremental* dengan pendekatan arsitektur *Model-View-Controller* (MVC) untuk memecah komponen aplikasi menjadi bagian-bagian yang dapat dikelola dengan lebih mudah.

Kata kunci: VBA Excel, Aplikasi Pengecekan Dokumen, *Iterative and Incremental Development*, Arsitektur MVC

I. PENDAHULUAN

Pesatnya arus perdagangan Internasional maupun domestik yang ada di Indonesia tidak hanya melibatkan lembaga-lembaga terkait saja, namun melibatkan juga perusahaan

yang bergerak di bidang jasa pelayanan pengiriman, baik dari keterlibatan dalam mempersiapkan dokumen-dokumen ekspor-impor maupun peti kemasnya (*container*) (Ginting & Ginting, 2021). Pengurusan dokumen barang ekspor merupakan suatu tahapan yang sangat penting dalam kegiatan ekspor barang, tanpa adanya dokumen kegiatan arus barang tidak akan berjalan sebagaimana mestinya. Dokumen ekspor juga

* firdaroziana10@gmail.com

Diterima: 20 Juni 2023

Direvisi: 3 Mei 2024

Disetujui: 7 Mei 2024

DOI: [10.23969/infomatek.v26i1.8680](https://doi.org/10.23969/infomatek.v26i1.8680)

digunakan sebagai alat bukti termasuk juga transaksi perdagangan luar negeri antar negara (Ginting & Ginting, 2021). Perusahaan yang bergerak di industri manufaktur yang memproduksi barang umumnya memiliki Departemen Ekspor sebagai penanggung jawab yang mengatur kegiatan perdagangan antar negeri tersebut. Seperti halnya pada Departemen Ekspor di PT. XY dalam tupoksinya adalah untuk melaksanakan proses alur ekspor barang dengan melibatkan proses dan persyaratan dokumentasi ekspor yang kompleks.

Dokumen ekspor berisi kumpulan dokumen yang diperlukan untuk mengekspor barang dari satu negara ke negara lain agar kegiatan ekspor tersebut sah dan legal. Penggunaan dokumen-dokumen ekspor tergantung dari jenis produk atau barang yang diekspor serta aturan perdagangan yang berlaku di masing-masing negara mitra dagang (Beatrix & Tewu, 2022). Pada Departemen Ekspor PT.XY beberapa dokumen ekspor yang dibutuhkan diantaranya adalah Bill of Lading (B/L) yaitu surat tanda terima barang, Certificate of Origin (CO) atau surat keterangan asal, dan Pemberitahuan Ekspor Barang (PEB). Ketiga dokumen ini masing-masing dikeluarkan oleh lembaga atau otoritas terkait dengan aturan dan prosedur yang berbeda. Departemen Ekspor bertanggung jawab untuk memverifikasi dan memastikan bahwa semua bagian dokumen yang diperlukan tersedia. Setiap dokumen yang telah diterbitkan oleh pihak otoritas akan disimpan dalam sebuah private FTP server yang hanya dapat diakses oleh Pegawai Departemen Ekspor untuk keperluan administrasi lanjut.

Untuk memastikan kelengkapan dokumen, pengecekan manual secara rutin dilakukan setiap hari pada setiap folder terpisah yang berisi dokumen BL, CO, dan PEB. Meskipun

pengecekan manual dilakukan dengan penuh ketelitian, namun pengecekan dengan cara ini dirasa tidak efektif dilakukan karena volume dokumen yang terus bertambah setiap harinya, yang membuat proses pengecekan memakan waktu yang cukup lama dan rentan terhadap kemungkinan kesalahan manusia, seperti melewatkan dokumen yang harus diperiksa atau kurang teliti dalam mencari informasi yang dibutuhkan. Oleh karena itu, dibutuhkan solusi yang lebih otomatisasi dan efisien untuk mempermudah akses dan pengecekan dokumen secara cepat dan akurat.

Berdasarkan latar belakang masalah tersebut, maka dibuatlah aplikasi pengecekan kelengkapan dokumen ekspor berbasis VBA Excel yang dapat mempermudah proses pengecekan dokumen secara praktis sehingga lebih efisien waktu. Macro merupakan fitur dalam Microsoft Excel yang dapat digunakan menjadi serangkaian perintah atau instruksi untuk melakukan otomatisasi tugas. Otomatisasi macro dibuat menggunakan bahasa pemrograman Visual Basic for Application (VBA). Sistem informasi yang dibuat menggunakan Macro VBA Excel akan mempermudah dokumentasi data sehingga informasi akan menjadi lebih cepat, terpadu dan mudah diperoleh dibandingkan dengan sistem manual (Lestari et al., n.d.).

Beberapa peneliti telah melakukan penelitian mengenai pengembangan aplikasi berbasis VBA Excel, seperti pada penelitian oleh Rosalin dan Ambarwati yang membangun aplikasi agenda surat berbasis VBA Excel dengan terintegrasi ke dalam database Microsoft Excel untuk mempermudah dan mempercepat pengelolaan agenda surat menyurat. Dalam penelitian tersebut, metode pengembangan sistem yang digunakan adalah metode *waterfall* dan metode rancang

bangun untuk melakukan pengamatan, analisis kebutuhan, dan desain sistem (Rosalin1 et al., n.d.). Penelitian lainnya dilakukan oleh Ilham dan Agung yang membuat aplikasi berbasis VBA Excel untuk pengelolaan dan manajemen stok barang gudang. Penelitian tersebut juga menerapkan tahapan pada metode pengembangan *waterfall* mulai dari analisis hingga pengujian sistem untuk pengembangan aplikasi yang dibuat (Herdaswara Fernando Naufal et al., 2020). Hal ini berbeda dengan penelitian yang hendak dilakukan oleh penulis, dimana penulis membuat aplikasi berbasis VBA Excel menggunakan metode *Iterative and Incremental* dengan pendekatan arsitektur MVC untuk mengelola dan memodifikasi komponen-komponen aplikasi secara terpisah, sehingga meningkatkan fleksibilitas dan mempercepat pengembangan aplikasi.

II. METODOLOGI

2.1. Teknik Pengumpulan Data

Pengumpulan data dilakukan untuk memperoleh hasil akhir yang sesuai dengan kebutuhan agar aplikasi dapat berjalan secara optimal. Teknik pengumpulan data yang dilakukan meliputi:

1. Observasi, melakukan pengamatan yaitu dengan cara mengamati aktivitas pengecekan secara manual yang sedang berlangsung dan untuk mengambil sampel data yang diperlukan dalam pembuatan aplikasi pengecekan kelengkapan dokumen.
2. Wawancara, melakukan sesi tanya jawab kepada calon pengguna aplikasi yaitu dengan Pegawai Departemen Ekspor untuk memperoleh data kebutuhan fungsionalnya.

2.2. Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan adalah model *Iterative and Incremental*. Model *Iterative and Incremental* adalah pengembangan sistem dengan siklus berulang yang melakukan iterasi pada setiap tahap dan mengevaluasi setelah satu tahap iterasi selesai. Hasil evaluasi tersebut akan digunakan sebagai bahan pengembangan pada iterasi selanjutnya (Maulanal et al., n.d.). Pengembangan *Iterative and Incremental* terdiri dari empat bagian yaitu (Lukitaningtyas et al., n.d.):

1. Tahap Insepsi (Permulaan)

Pada tahap ini memodelkan proses bisnis yang dibutuhkan dan mendefinisikan kebutuhan sistem yang akan dibuat.

2. Tahap elaborasi (Perencanaan)

Elaborasi memfokuskan pada pembuatan analisis dan desain level tinggi, menerbitkan arsitektur dasar untuk proyek yang akan dikerjakan, membuat rencana konstruksi yang mendukung pencapaian tujuan dari proyek.

3. Tahap Konstruksi

Tahap konstruksi fokus pada pengembangan komponen dan fitur-fitur sistem untuk menghasilkan prototype atau produk piranti lunak.

4. Tahap Transisi

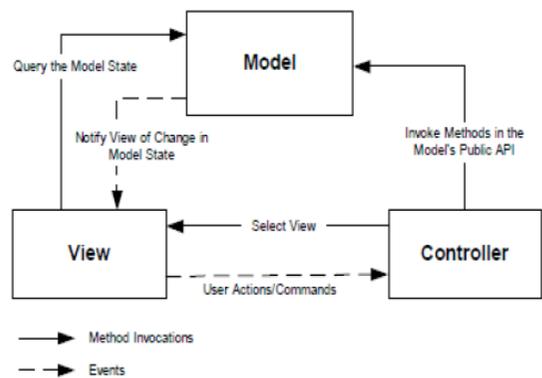
Tahapan transisi lebih menekan pada instalasi sistem agar dapat dimengerti oleh user. Pada akhir tahap transisi sistem akan dirilis dan proyek akan dievaluasi.

Model pengembangan *Iterative and Incremental* memiliki kelebihan antara lain dapat mengembangkan persyaratan yang diprioritaskan terlebih dahulu, *delivery* produk lebih cepat, fungsionalitas utama dari produk dapat diselesaikan terlebih dahulu, dan

perubahan pada persyaratan produk dapat dengan mudah ditampung (Bilad & Isyanto, n.d.). Dalam pembuatan aplikasi pengecekan dokumen ekspor ini pengembangan menggunakan pendekatan yang berorientasi kepada objek dengan arsitektur MVC (*Model-View-Controller*). Menurut Deacon (2009), "MVC adalah sebuah konsep yang diperkenalkan oleh Smalltalk (Trygve Reenskaug) untuk meng-*enkapsulasi* data bersama dengan pemrosesan (*model*), mengisolasi dari proses manipulasi (*controller*) dan tampilan (*view*) untuk direpresentasikan pada tampilan antarmuka." (Hidayat & Surarso, 2012). Definisi teknis dari arsitektur MVC dibagi menjadi tiga lapisan (Hidayat & Surarso, 2012):

- a. *Model*, digunakan untuk mengelola informasi dan memberitahu pengamat ketika ada perubahan informasi. Hanya model yang mengandung data dan fungsi-fungsi yang berhubungan dengan pemrosesan data.
- b. *View*, bertanggung jawab untuk pemetaan grafis ke sebuah perangkat. View melekat pada model dan me-render isinya ke permukaan layar.
- c. *Controller*, menerima input dari pengguna dan menginstruksikan model dan *view* untuk melakukan aksi berdasarkan masukan tersebut. Sehingga, *controller* bertanggung jawab untuk pemetaan aksi pengguna akhir terhadap respon aplikasi.

Model, *view* dan *controller* sangat erat terkait, oleh karena itu, mereka harus merujuk satu sama lain. Gambar 1. mengilustrasikan hubungan dasar *Model-View-Controller* (Hidayat & Surarso, 2012).



Gambar 1. Hubungan antara *model*, *view*, dan *controller* (Gulzar, 2002)

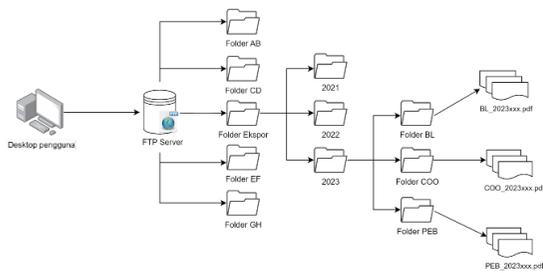
Arsitektur MVC memiliki manfaat yaitu menjadi pemisahan antara *model* dan *view* memungkinkan beberapa *view* menggunakan *model* yang sama. Akibatnya, komponen *model* sebuah aplikasi lebih mudah untuk diterapkan, diuji, dan dipelihara, karena semua akses ke model berjalan melalui komponen ini (Hidayat & Surarso, 2012).

III. HASIL DAN PEMBAHASAN

3.1 Analisis Perancangan Sistem

a. Metode Pengecekan Saat Ini

Hasil observasi penulis terhadap proses pengecekan dokumen ekspor yang dilakukan secara manual adalah Pegawai perlu masuk ke FTP server yang sudah tertanam pada *personal computer*, kemudian membuka folder khusus dokumen ekspor yang didalamnya terdapat folder dokumen BL, folder dokumen CO, dan folder dokumen PEB. Pegawai melakukan pengecekan satu per satu untuk memastikan setiap dokumen berdasarkan invoice tertentu ada di dalam foldernya masing-masing. Gambar 2 berikut ini menunjukkan skema folder di dalam FTP server



Gambar 2. Letak folder dokumen ekspor pada FTP server Departemen Ekspor

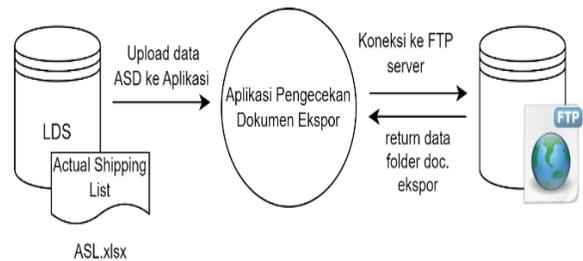
Pengecekan yang dilakukan secara manual memerlukan ketelitian pemeriksaan yang memakan waktu cukup lama karena Pegawai harus membuka beberapa folder terpisah dan mencari dokumen berdasarkan invoice yang ingin dicari. Cara ini tidak efektif dilakukan karena jumlah dokumen akan terus bertambah setiap harinya. Selain itu Pegawai juga tidak dapat langsung memeriksa dokumen dalam jumlah banyak jika invoice yang ingin dicari lebih dari satu invoice.

b. Sistem yang Diusulkan

Berdasarkan hasil observasi dan wawancara yang dilakukan penulis dengan Pegawai ekspor, penulis membuat rancangan dan mengembangkan sebuah aplikasi berbasis VBA Excel yang dapat melakukan pengecekan kelengkapan dokumen ekspor secara otomatis dan terintegrasi dengan FTP server. Gambar 3 merupakan rancangan skema pengecekan dokumen ekspor dari sistem aplikasi yang diusulkan.

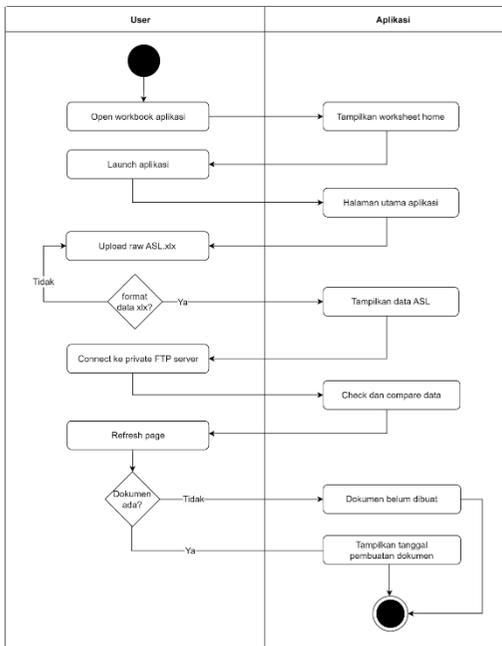
Untuk dapat memeriksa dokumen secara keseluruhan tanpa harus mencari satu per satu invoice, Pegawai perlu mengunduh data *Actual Shipping List* dari *Logistic Document Module* (LDS) dengan rentang waktu pada tanggal awal dan akhir bulan saat ini. *Actual Shipping List* berisi informasi daftar *invoice*

yang berjalan di bulan ini, termasuk dengan jenis barangnya, jumlah, berat, dimensi, nomor dan tipe kontainer. Data *Actual Shipping List* yang berformat *xlsx*. diunggah ke dalam aplikasi dan tersimpan ke dalam worksheet *temporary database*.



Gambar 3. Skenario sistem usulan

Aplikasi diintegrasikan dengan FTP server, sehingga ketika terhubung secara otomatis akan mendapatkan lokasi folder BL, folder CO, dan folder PEB sesuai tahun dan bulan saat ini. Setelah mendapatkan lokasi folder, macro akan mencetak nama dan tanggal kapan dokumen BL, dokumen CO, dan dokumen PEB disimpan di dalam folder tersebut. Nama dan tanggal dokumen yang dicetak dimasukkan ke dalam *temporary database* aplikasi sebagai bahan perbandingan untuk mencocokkan data dengan daftar nomor *invoice* yang berasal dari *Actual Shipping List* yang sebelumnya telah diunggah ke dalam sistem. Dari alur sistem yang diusulkan maka proses yang terjadi pada saat melakukan pengecekan dokumen melalui aplikasi yang dibuat mengikuti pola seperti pada activity diagram pada Gambar 4.



Gambar 4. Activity diagram alur proses kejadian pengguna melakukan pengecekan dokumen

c. Kebutuhan Sistem

Dari skema sistem yang diusulkan, kebutuhan sistem secara fungsional pada aplikasi pengecekan dokumen ekspor adalah sebagai berikut:

1. Pengguna harus menyiapkan raw data dari *Actual Shipping List* yang diunduh dari LDS untuk dapat mengambil daftar invoice yang sudah dibuat pada bulan ini.
2. Pengguna mengupload *Actual Shipping List* kedalam database aplikasi
3. Aplikasi dapat terkoneksi dengan private FTP server untuk mengambil folder-folder dokumen ekspor
4. Pengguna menginput nomor invoice jika ingin mencari sebuah invoice

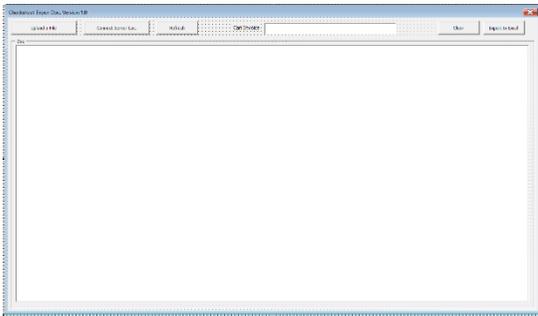
3.2 Implementasi Arsitektur MVC (Model-View-Controller)

a. Model

Model merupakan representasi dari proses bisnis di dalam setiap aplikasi perangkat lunak. *Model* bertugas mengolah data mentah menjadi data yang mengandung arti yang diinginkan oleh pengguna. *Model* biasanya berhubungan langsung dengan basis data untuk memanipulasi data, menangani validasi dari bagian *controller*, namun tidak dapat berhubungan langsung dengan bagian *view* (Dwi Praba STMIK Nusa Mandiri Jakarta, 2018). Pada perancangan aplikasi ini, salah satu aspek yang melibatkan pemrosesan dalam model adalah pengolahan data pada worksheet yang difungsikan sebagai *temporary database* atau sebagai wadah sementara untuk manipulasi data sebelum ditampilkan ke *view*. Menggunakan worksheet penyimpanan sementara ini, aplikasi dapat menyimpan data yang diunggah oleh pengguna dan data yang diambil dari server untuk dimanipulasi sesuai dengan rumus yang diterapkan dalam worksheet. *Model* memastikan aplikasi dapat mengirim permintaan data dan menerima respons dari server dengan baik. Skema penyimpanan sementara ini menjadi bagian dari proses *back-end* aplikasi untuk mengelola data yang diperlukan saat proses pengecekan dokumen, sehingga worksheet ini disembunyikan karena tidak perlu ditampilkan kepada pengguna.

b. View

View menjadi bagian yang diperlukan untuk menampilkan *interface* aplikasi dan informasi yang dibutuhkan. Aplikasi yang dibuat hanya memiliki satu userForm sebagai tampilan GUI dengan beberapa elemen *user interface* (UI) di dalamnya seperti *action button*, *listBox* untuk menampilkan data, *label*, dan *inputBox*. Gambar 5 memperlihatkan komponen-komponen yang terdapat di userForm



Gambar 5. userForm dari Aplikasi

userForm dipanggil melalui *lunch button* yang berisi perintah macro untuk membuka form. *Lunch button* terdapat di worksheet utama, yaitu worksheet GUI. Pada Gambar 6 dan Gambar 7 terdapat bagian dari code VBA yang digunakan untuk menampilkan userForm dan mengatur tampilan kolom *listBox* untuk menampung data *Actual Shipping list* yang telah diunggah.

```
Sub Show_Form()
Dim sh As Worksheet

For Each sh In ThisWorkbook.Sheets
If sh.Name = "GUI" Then
sh.Visible = xlSheetVisible
Else
sh.Visible = xlSheetHidden
End If
Next
GUI_UserForm.Show False

End Sub
```

Gambar 6. Kode memanggil userForm

```
Sub data ()

With GUI_UserForm.ListDatabase
.ColumnCount = 6
.ColumnHeads = False

.ColumnWidths = "25;60;30;120;70;"
.RowSource = "data_from_upload"

.MultiSelect = fmMultiSelectSingle
.BoundColumn = 0

End With

End Sub
```

Gambar 7. Kode untuk mengatur tampilan dan ukuran kolom dari data ASL

Sumber data untuk kolom ASL diambil melalui properti *.RowSource* dengan merujuk pada rentang cells bernama "data_from_upload" yang sumber datanya tersimpan di *temporary database*. Setelah data terupload, aplikasi dihubungkan ke FTP server untuk mengambil data dari folder dokumen ekspor. Data yang berhasil diunggah dimasukkan ke dalam *temporary database* dan diolah untuk proses pengecekan dokumen. Aplikasi kemudian di-*refresh* menggunakan *refresh button* dan ditampilkan ke dalam *listBox*. Pada gambar 8 terdapat bagian dari *code* untuk me-*refresh* konten *listBox*

```
Sub refresh()

With GUI_UserForm.ListDatabase
.ColumnCount = 9
.ColumnHeads = False
.ColumnWidths = "25;60;30;120;70;65;60;60;60;"
.RowSource = "final_result"
.MultiSelect = fmMultiSelectSingle
.BoundColumn = 0

End With

End Sub

Private Sub btnRefresh_click()
Call refresh

End Sub
```

Gambar 8. Kode untuk me-*refresh* data dalam *listBox* dan memperbarui layout sesuai dengan ukuran lebar kolom baru

Terdapat sembilan kolom pada *listBox* dengan enam kolom untuk menampilkan informasi dari *Actual Shipping List* (termasuk nomor *invoice*) dan tiga kolom untuk menampilkan tanggal pembuatan dokumen BL, CO, dan PEB.

c. Controller

Controller bertanggung jawab sebagai penghubung antara *model* dan *view* untuk mengatur interaksi keduanya. *Controller* menerima input dari pengguna melalui interaksi pada *view*, kemudian *controller* mengambil data dari *model* dan meneruskannya ke *view* untuk ditampilkan

pada userForm. Seperti tertera pada Gambar 9 dan Gambar 10 yang menunjukkan potongan kode program untuk mencari dan mengatur pengambilan informasi dari FTP server yang dihubungkan ke aplikasi.

```
Sub get_BL_Doc()

'get based on bulan dan tahun saat ini
Dim bulan, tahun As Integer
bulan = Month(Date)
tahun = Year(Date)

'inisiasi nama folder bulan & tahun
Dim nama_folder As String
nama_folder = Format(Date, "MMMM YYYY")

'path/lokasi folder
Dim path_folder As String
path_folder = "D:\Server X\Ekspor\" _
& nama_folder & "\BL"

If Dir(path_folder, vbDirectory) <> "" Then
'get doc.pdf
    Dim file_name As String
    file_name = Dir(path_folder & "\*.pdf")
    If file_name <> "" Then
        'direct ws database
        Dim current_ws As Worksheet
        Dim current_row As Integer
        Set current_ws = Worksheets("database")
        current_row = 2

        'Print nama & tgl kapan dokumen diletakan
        Do While file_name <> ""
            Dim file_path As String
            file_path = path_folder & "\" & file_name

            Dim fileDateDoc As String
            fileDateDoc = Format(FileDateTime(file_path), _
"dd/mm/yyyy")
            'save ke kolom J (col. untuk menampung BL
            current_ws.Cells(current_row, 10).Value = _
file_name & "-" & fileDateDoc
            current_row = current_row + 1
            file_name = Dir()
            Loop
        Else
            MsgBox "Tidak ada file BL di dalam folder."
        End If
    Else
        MsgBox "Folder tidak ditemukan."
    End If
End If
```

Gambar 10. Kode untuk mengambil informasi dokumen BL

Ketika *connect button* pada UserForm di klik, *controller* akan mengambil data pada FTP server melalui macro sub `get_BL_doc`

berdasarkan alamat direktori folder tersebut. Data yang telah diambil diteruskan ke dalam *temporary database* untuk diproses. Hasil akhir pengecekan diupdate kembali ke dalam *listBox* dengan memperbarui isi tampilannya. Kode program yang sama seperti sub `get_BL_doc` dibuat juga untuk pengambilan data dokumen PEB dan dokumen CO, yang membedakan adalah *path* direktori dan format penamaannya. Gambar 11 menunjukkan komponen-komponen sub yang dipanggil untuk terhubung ke FTP server

```
Sub connect_server_local()
    Call get_BL_Doc
    Call get_CO_doc
    Call get_PEB_doc
End Sub

Private Sub btnConnect_click()
    Call connect_server_local
    MsgBox _
("Koneksi ke server local berhasil!"), _
vbInformation
End Sub
```

Gambar 11. Kode komponen sub untuk terhubung ke FTP server

Pada Gambar 11 komponen sub dikelompokkan ke dalam sub `connect_server_local` dan dipasang pada *connect button* sehingga dapat melakukan *action* pada direktori folder yang berbeda hanya dengan satu kali panggilan.

3.3 Implementasi Desain Aplikasi

Aplikasi memiliki dua worksheet yang terbagi menjadi worksheet GUI dan worksheet temporary database. GUI berfungsi sebagai bagian *front-end* yang akan digunakan sebagai interface pengguna, sedangkan temporary database menjadi bagian *back-end* untuk mengolah data. Berikut adalah detail tampilan aplikasi pengecekan dokumen ekspor berbasis VBA Excel

a. Tampilan worksheet

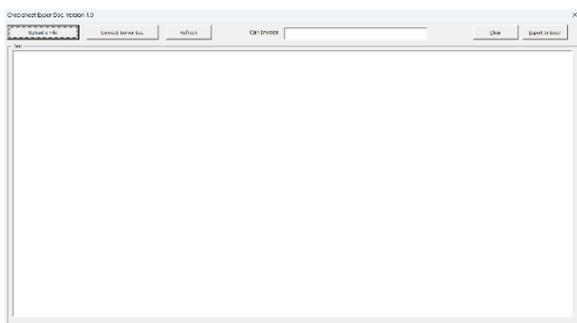
Worksheet GUI pada Workbook aplikasi pengecekan dokumen ekspor. Worksheet ini merupakan halaman utama ketika pengguna membuka workbook aplikasi. Seperti pada Gambar 12, di dalam worksheet terdapat *launch button* untuk menjalankan aplikasi.



Gambar 12. Tampilan Worksheet: GUI

a. Tampilan userForm

Secara sederhana, pengguna dapat mengupload *Actual Shipping List* sebagai material awal untuk melakukan pengecekan dokumen dan menghubungkan aplikasi ke server melalui *form* ini. Seluruh tahapan fungsional pada proses pengecekan (termasuk fitur pencarian *invoice* dan *export* hasil pengecekan kedalam Excel baru) dilakukan secara langsung pada *form* ini. Gambar 13 menunjukkan tampilan userForm yang menjadi GUI utama pada aplikasi



Gambar 13. userForm aplikasi pengecekan dokumen ekspor

Data *Actual Shipping List* yang telah diunggah akan ditampilkan ke dalam userForm. Seperti pada Gambar 14, display menampilkan informasi berisi nomor *invoice*, destinasi pengiriman, negara asal *buyer*, ETD (perkiraan waktu keberangkatan) dan keterangan TWH (*warehouse*).

No	Invoice No	Dest.	Buyer	ETD	TWH
2	22B8SD1	PZS	PORT SUDAN, SUDAN	03/20/2023	Karawang
3	22B23NC1A	NOU	NOUMEA, NEW CALEDONIA	03/31/2023	Karawang
4	22J4A4	TYA	TOKYO, JAPAN	03/06/2023	Karawang
5	22C23A	SWI	SENDAL, JAPAN	03/02/2023	Karawang
6	22C23B	SWI	SENDAL, JAPAN	03/03/2023	Bekasi
7	22D53A	SND	SENDAL, JAPAN	03/02/2023	Karawang
8	22D04A	SMZ	SHIMIZU, JAPAN	03/01/2023	Karawang
9	22J23A	HD	HIROSHIMA, JAPAN	03/01/2023	Karawang
10	22J23A	HD	HIROSHIMA, JAPAN	03/01/2023	Bekasi

Gambar 14. userForm dengan data *Actual Shipping List* yang telah diunggah ke dalam aplikasi

b. Tampilan userForm dengan hasil pengecekan

Setelah mengupload data *Actual Shipping List*, pengguna dapat menghubungkan aplikasi ke FTP server melalui *connect button*. Apabila dokumen ekspor (dokumen BL, CO dan PEB) yang diperlukan telah ada di folder FTP server, maka aplikasi akan menampilkan tanggal publishnya dokumen tersebut sesuai nomor *invoice* yang berkaitan. Gambar 15 menunjukkan hasil simulasi pengecekan berdasarkan nomor *invoice* yang telah memiliki dokumen yang dipersyaratkan.

ETD	TWH	B/L Doc.	CO Doc.	PEB Doc.
03/20/2023	Karawang	28/03/2023	28/03/2023	28/03/2023
03/31/2023	Karawang	28/03/2023		
03/06/2023	Karawang			28/03/2023
03/02/2023	Karawang	28/03/2023		28/03/2023
03/03/2023	Bekasi		28/03/2023	
03/02/2023	Karawang			
03/01/2023	Karawang			
03/01/2023	Karawang		28/03/2023	

Gambar 15. Hasil pengecekan pada aplikasi

c. Tampilan Fitur Pencarian Invoice

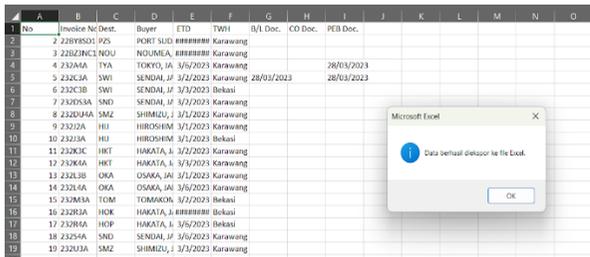
Aplikasi dapat melakukan pencarian berdasarkan nomor *invoice* menggunakan fitur pencarian yang tersedia pada *toolbar* aplikasi. Hasil pencarian akan disorot dan ditempatkan di bagian atas *listBox* seperti pada Gambar 16



Gambar 16. Fitur pencarian berdasarkan nomor *invoice* pada aplikasi

d. Tampilan Export data ke Excel

Setelah proses pengecekan, pengguna dapat mengekspor hasil pengecekan sebagai bahan reporting untuk keperluan *internal department* jika dibutuhkan. Fitur export to excel akan mencetak data dengan membuat *workbook* excel baru yang menampung informasi hasil pengecekan dokumen. Gambar 17 menunjukkan hasil dari fitur exporting



Gambar 17. Fitur Export data ke Excel

e. Tampilan Worksheet Database (Penyimpanan Sementara yang tersembunyi)

Selain Worksheet untuk menampung laman GUI, aplikasi juga memiliki *worksheet* lain yang disembunyikan sebagai tempat penyimpanan atau basis data sementara. Pada Gambar 18 terlihat bahwa worksheet ini memiliki satu tabel dengan beberapa kolom didalamnya. Kolom pertama merupakan kolom dalam range A:F untuk menampung hasil upload data Actual Shipping List. Kolom kedua yaitu pada range G:I, merupakan hasil pengecekan dokumen menggunakan rumus excel yang mengandung perintah IFERROR, MID, Index Match dan FIND. Kolom ketiga,

range J:L merupakan kolom untuk menampung hasil penarikan informasi data dari folder dokumen ekspor di FTP server



Gambar 18. Worksheet database untuk menampung dan memanipulasi data

IV. KESIMPULAN

Berdasarkan hasil analisis, perancangan sistem dan implementasi pengembangan Aplikasi Pengecekan Dokumen Ekspor yang dilakukan, maka penulis dapat menyimpulkan sebagai berikut:

1. Pengembangan aplikasi berbasis VBA memanfaatkan Microsoft Excel sebagai lingkup platform yang digunakan. Aplikasi ini menjadi lebih familiar untuk diimplementasikan karena tidak memerlukan *requirement* dan serangkaian alur yang kompleks untuk menjalankan fungsionalnya.
2. Aplikasi Pengecekan Dokumen Ekspor ini dibuat berbasis Visual Basic Application dan Excel Macro menggunakan model pengembangan sistem *Iterative and Incremental* dengan implementasi pendekatan MVC (*Model-View-Controller*)
3. Aplikasi terintegrasi dengan private FTP server milik Departemen Ekspor, sehingga dapat menghasilkan pemeriksaan yang akurat dan valid terkait pengambilan informasi dokumen
4. Dengan menggunakan aplikasi ini untuk mengecek kelengkapan dokumen ekspor, Departemen Ekspor PT. XY dapat dengan

mudah melakukan proses pengecekan dokumen, sehingga tidak perlu mencari secara manual invoice mana saja yang belum memiliki komponen dokumen ekspor yang diperlukan (Dokumen BL, CO, atau PEB)

5. Aplikasi Pengecekan Dokumen Ekspor ini dapat dikembangkan lagi agar lebih maksimal dengan menambahkan fitur lain seperti *auto filter* untuk mengurutkan data berdasarkan abjad urutan invoice, atau kategori tertentu. Aplikasi juga dapat ditambahkan kolom yang menunjukkan status akhir keterangan dokumen dalam bentuk *mark* atau *checkpoint* sehingga lebih mudah mempresentasikan data yang kurang lengkap dalam bentuk visual kepada pengguna aplikasi.

DAFTAR PUSTAKA

- Beatrix, S. G. S., & Tewu, M. L. D. (2022). Analisis Resiko Ekspor Bola Kaki ke Brazil. *JOURNAL RESPECS*, 4(2), 44–55.
<https://doi.org/10.31949/respecs.v4i2.2629>
- Bilad, A., & Isyanto, G. (n.d.). Perancangan Back Framework Php Menggunakan Metode Iterative Dan Building Back-End Application On Warung Finder With Php Framework Using.
- Dwi P, STMIK Nusa Mandiri Jakarta, A. (2018). Implementasi Model View Controller Dengan framework CodeIgniter Pada Perpustakaan. *Ijse.Web.Id IJSE-Indonesian Journal on Software Engineering*, 4(1).
- Ginting, D., & Ginting, D. G. (2021). Perananan Keagenan Kapal Dalam Melayani Pengisian Air Bersih Untuk Kebutuhan Km.Amrta Vii Pada Pt. Gesuri Lioyd Cabang Kuala Tanjung. *Journal of Maritime and Education (JME)*, 3(2), 245–249.
<https://doi.org/10.54196/jme.v3i2.47>
- Herdaswara, P, N, I., Handayanto, A., Pusat Lantai, G., & Sidodadi T, J. (2020). Aplikasi Stok Barang Gudang Berbasis Vba Excel Dengan Metode Waterfall Di Upttik Upgris. In *Science And Engineering National Seminar* (Vol. 5).
- Hidayat, A., & Surarso, B. (2012). Penerapan Arsitektur Model View Controller (Mvc) Dalam Rancang Bangun Sistem Kuis Online Adaptif. *Seminar Nasional Teknologi Informasi Dan Komunikasi*.
- Lestari, T., Putri, K., Santoso, B., Choiri, M., Kunci, K., Sistem, :, Manajemen, I., Dan, K., & Kerja, K. (n.d.). *HEALTH & Safety Environment Management Information System Design Using Microsoft Excel 2007 And Vba Macro (Case Study: Pt.Beiersdorf Indonesia Malang)*.
- Lukitaningtyas, I., Andreswari, R., Al-Anshary, F. M., S1, P., Informasi, S., Industri, R., & Telkom, U. (n.d.). Rancang Bangun E-Marketplace “Dyland” Bagi Penyedia Jasa Event Organizer-Party Planner Menggunakan Metode Iterative Incremental (Modul Transaksi) Architecture Of E-Marketplace “Dyland” For Event Organizer-Party Planner Provider Using Iterative Incremental Method (Transaction Module).
- Maulana, M. F., Ramadani, L., & Al-Anshary, F. M. (n.d.). Pengembangan Sistem Telemedicine Berbasis Aplikasi Mobile Menggunakan Metode Iterative Dan Incremental Development Of A Telemedicine System Based On Mobile Applications Using Itrative And Incremental Methods.

Rosalin, S., Sri, D., & Ambarwati, A. (n.d.).
Perancangan Dan Pembuatan Aplikasi
Agenda Surat Berbasis Visual Basic For
Application (Vba) Excel Di Dinas
Pemberdayaan Masyarakat Dan Desa

Kabupaten Lumajang *Abstract:*
Designing and Creating Letter Agenda
Application Using VBA (Visual Basic for
Application) Excel in Dinas
Pemberdayaan.