

# Pengembangan Modul *Stock Fulfillment Online Transaction* pada Aplikasi *Distribution Center System* menggunakan Framework Flask di PT.XYZ

Tifano Eng Ogotan <sup>1</sup>, Felix David <sup>2\*</sup>

<sup>1,2\*</sup> Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Kota Salatiga, Provinsi Jawa Tengah, Indonesia.

*Email:* 672020019@student.uksw.edu <sup>1</sup>, felix@uksw.edu <sup>2\*</sup>

## Histori Artikel:

*Dikirim* 28 Februari 2024; *Diterima dalam bentuk revisi* 2 April 2024; *Diterima* 11 April 2024; *Diterbitkan* 10 Mei 2024. Semua hak dilindungi oleh Lembaga Penelitian dan Pengabdian Masyarakat (LPPM) STMKI Indonesia Banda Aceh.

## Abstrak

Semakin pesatnya kemajuan teknologi informasi yang terus berkembang, setiap perusahaan bisnis perlu selalu menyesuaikan diri agar sejalan dengan kemajuan teknologi. PT. XYZ adalah salah satu perusahaan bisnis yang berjalan pada bagian retail dan distribusi. PT.XYZ mempunyai masalah dalam teknologi lama yang digunakan yaitu Oracle Form. Oracle Form mempunyai kekurangan pada biaya pemeliharaan dan juga hanya dapat berjalan pada teknologi lama. Karena itu, PT.XYZ memutuskan untuk membuat modul Stock Fulfillment Online Transaction menggunakan framework Flask dan Python. Dalam pengembangan modul Stock Fulfillment Online Transaction menggunakan metode Agile Scrum. Tahap penelitian mencakup analisa kebutuhan, perencanaan sprint, pengembangan iterative, pengujian terus-menerus dan evaluasi akhir. Hasil dari pengembangan aplikasi ini menunjukkan bahwa aplikasi dengan menggunakan framework Flask dan Python dapat menghasilkan aplikasi yang bisa dioperasikan pada banyak platform dan juga pengurangan biaya pemeliharaan.

**Kata Kunci:** Flask; Python; Stock Fulfillment; Agile; Scrum.

## Abstract

The rapid advancement of information technology necessitates that every business adapts to keep pace with technological developments. PT. XYZ, a retail and distribution company, faced challenges with their outdated technology, specifically Oracle Forms. Oracle Forms incurred high maintenance costs and was limited to running on legacy systems. Consequently, PT.XYZ decided to create a Stock Fulfillment Online Transaction module using the Flask frameworks and Python. The development process followed the Agile Scrum methodology, including stages such as requirement analysis, sprint planning, iterative development, continuous testing, and final evaluation. The outcome of this application development demonstrates that using Flask frameworks and Python allows for an application that can operate across various platforms while reducing maintenance costs.

**Keyword:** Flask; Python; Stock Fulfillment; Agile; Scrum.

## 1. Pendahuluan

Semakin pesatnya kemajuan teknologi informasi yang terus berkembang, setiap perusahaan bisnis perlu selalu menyesuaikan diri agar sejalan dengan kemajuan teknologi (Anjani, 2019). PT. XYZ merupakan satu dari banyak perusahaan bisnis yang bergerak pada bidang retail dan distribusi. Retail umumnya merupakan aktivitas komersial yang menjual berbagai jasa dan barang untuk konsumsi langsung maupun tidak langsung. Pada rantai perdagangan, eceran merupakan bagian akhir dari pendistribusian jasa dan barang yang berhadapan langsung dengan pembeli (Lapian, 2017). PT. XYZ semakin berkembang sebagai perusahaan retail dan distribusi terbesar yang memiliki belasan ribu toko tersebar di seluruh Indonesia dan sudah melakukan jutaan transaksi setiap harinya. Transaksi yang sebelumnya hanya berfokus pada penjualan secara konvensional kini semakin memperluas bisnis perusahaan dengan mengadopsi sistem penjualan secara online (Octaviani *et al.*, 2022). Hal ini tidak hanya memberikan keunggulan dan kenyamanan bagi pelanggan, tetapi juga menjadi peluang bagi perusahaan untuk menyediakan layanan online sebagai pendukung dari layanan konvensional yang ditawarkan.

Saat ini PT.XYZ memiliki sebuah aplikasi perangkat lunak digital yang menyediakan beragam produk dan transaksi secara online (Alfagift, 2023). Meskipun aplikasi ini menawarkan kemudahan dan kenyamanan, terdapat permasalahan ketika toko tidak memiliki persediaan barang yang mengakibatkan pelanggan tidak dapat melakukan pembelian. Hal ini dapat mempengaruhi pelanggan yang berbelanja melalui layanan online. Sehingga PT. XYZ mengambil langkah untuk membuat aplikasi lain yang mampu berintegrasi dengan aplikasi. Akan tetapi teknologi yang dipakai masih menggunakan Oracle Forms yang dinilai tidak efisien dan sudah tidak kompatibel dengan era sekarang ini (Microsoft Community, 2023). Sehingga perlu dilakukan pembaharuan aplikasi berbasis web menggunakan bahasa Python dengan database PostgreSQL dan framework Flask. Pengembangan Modul *Stock Fulfillment Online Transaction* digunakan untuk memudahkan proses transaksi online, apabila stok toko habis maka barang akan diambil langsung dari gudang. Sehingga proses bisnis tetap berlangsung. Ketersediaan stok barang dan juga kecepatan pengiriman sangat berpengaruh terhadap kepuasan pelanggan (Bramasta & Ikram, 2023). Kepuasan pelanggan dan kualitas pelayanan juga berpengaruh besar terhadap loyalitas pelanggan, sehingga pelanggan selalu berbelanja (Alwi *et al.*, 2021).

Pemilihan Flask dan python sebagai bahasa pemrograman untuk membangun Modul *Stock Fulfillment Online Transaction* karena landasan aplikasi dan antarmuka pengguna pada berbasis web. Dengan memakai framework Flask dan bahasa pemrograman Python, pengembang memiliki kemampuan untuk membuat web yang baik dan terstruktur serta mengelola web dengan lebih mudah (Irsyad, 2018). WSGI (*Web Server Gateway Interface*) adalah spesifikasi yang menggambarkan komunikasi antara server web dan aplikasi web Python atau kerangka kerja. WSGI menjelaskan bagaimana server web berkomunikasi dengan aplikasi web Python atau kerangka kerja, serta bagaimana aplikasi web atau kerangka kerja dapat dihubungkan untuk memproses permintaan (Ghimire, 2020). Python adalah salah satu bahasa pemrograman tingkat tinggi yang diciptakan untuk memberikan kenyamanan kepada para programmer melalui keefisienan waktu, pengembangan, kemudahan, dan kompatibilitas dengan system (Saragih, 2016). Python berjalan pada sistem interpreter, sehingga kode dapat dieksekusi segera setelah ditulis. Ini memungkinkan prototyping menjadi sangat cepat (Vyshnavi & Malik, 2019). PostgreSQL adalah Sistem Manajemen Basis Data Objek Relasional yang bersifat open source, artinya source code dari PostgreSQL dapat digunakan secara bebas. PostgreSQL memiliki dukungan terhadap *Structured Query Language* dengan kemampuan transaksi, subquery, dan fitur lainnya (Praba & Safitri, 2020).

Pada penelitian “Implementasi Flask pada Sistem Penentuan Minimal Order untuk Tiap Item Barang di Distribution Center Pada PT XYZ Berbasis Website” menjelaskan penggunaan metode WaterFall dalam pengembangan dan pengujian sistem menggunakan teknik black box dengan pengujian jaminan kualitas. Hasil penelitian menunjukkan bahwa sistem yang dibangun dapat mengatur jumlah minimum pemesanan setiap produk dengan baik. Saran untuk pengembangan

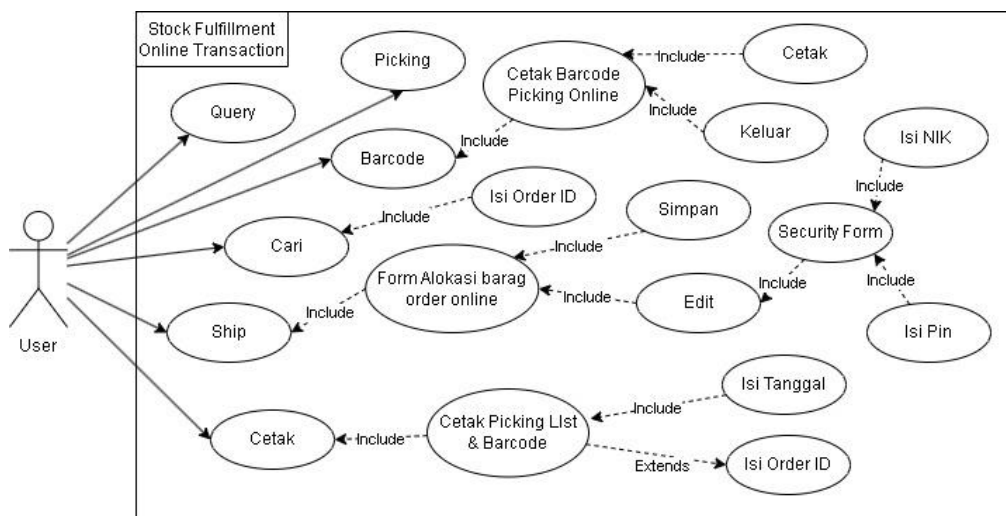
sistem mencakup penggunaan validasi Python dan peningkatan penanganan kesalahan (Novindri & Saian, 2022). Pada penelitian “Pengembangan Modul Inventory Management pada Aplikasi Master Distribution Centre System Menggunakan Framework Flask di PT XYZ” membahas tentang pengembangan modul manajemen inventaris pada aplikasi Master Distribution Centre System (MDCS) di PT XYZ1. Penelitian ini dilakukan dalam empat tahap, dari tahap analisis berlanjut ke tahap desain juga tahap pengkodean sampai tahap pengujian. Hasil dari penelitian ini adalah modul manajemen inventaris yang dibangun dengan framework Flask dan informasi pengembangan modul manajemen inventaris. Modul ini memenuhi persyaratan bisnis PT XYZ dan lulus pengujian black box dan pengujian penerimaan pengguna (Budianto & Saian, 2023).

Pada penelitian “Penerapan Framework Flask pada Pembangunan Sistem Informasi Pemasok Barang” membahas pengembangan sistem informasi pemasok barang pada perusahaan ritel. Penelitian ini dilakukan dalam beberapa tahap mulai dari analisis kebutuhan sistem masuk pada perencanaan juga desain sistem serta implementasi, dan pengujian sistem. Sistem informasi pemasok produk ini dibuat dalam format web dan PostgreSQL sebagai basis data. Hasil pengujian menunjukkan bahwa aplikasi yang diuji mendapatkan respon yang baik, dengan 96,67% pengguna setuju jika sistem yang dibuat sangat memudahkan dalam proses pengiriman barang (Walingkas & Saian, 2023).

## 2. Metode Penelitian

Dalam pengembangan Modul *Stock Fulfillment Online Transaction* menggunakan penerapan metode agile yang memberikan pendekatan adaptif dan responsif terhadap perubahan kebutuhan. Berikut metode penelitian dengan menggunakan pendekatan agile:

- 1) **Analisa Kebutuhan**  
Analisa kebutuhan merupakan langkah awal untuk merinci kebutuhan *fungsiional* dan *non-fungsiional* dalam mengembangkan modul yang akan dikerjakan.
- 2) **Perencanaan *Sprint***  
Perencanaan *Sprint* merupakan proses membagi pengembangan menjadi *sprint* yang terfokus yang biasa pengerjaannya dalam dua atau empat minggu.
- 3) **Pengembangan *Iterative***  
Pengembangan *Iterative* merupakan suatu pendekatan dalam pengembangan dimana project dibagi menjadi serangkaian siklus pendek. Setiap iterasi menghasilkan versi produk yang dapat diuji dan dievaluasi, serta mencakup tahap perencanaan, analisis, desain, implementasi, pengujian dan evaluasi. Pendekatan ini dapat membuat pengembang dapat merespon perubahan kebutuhan atau masalah yang muncul saat proses pengembangan dengan cepat.
- 4) **Pengujian Terus-menerus**  
Pengujian Terus-menerus merupakan proses yang dilakukan tim QA untuk memastikan bahwa Modul yang dibuat sudah sesuai dengan yang diinginkan.
- 5) **Evaluasi Akhir**  
Evaluasi akhir merupakan Identifikasi pelajaran yang dapat diambil dan pengembangan untuk perbaikan di masa mendatang.



Gambar 1. Use Case Diagram

Gambar 1 adalah *use case* diagram dari modul *Stock Fulfillment Online Transaction* yang menjelaskan hal yang dapat dilakukan user terhadap aplikasi. *User* dapat mengambil data dan di tampilkan, dapat melakukan proses picking, hingga ship. *User* juga dapat melakukan pencetakan, yaitu dengan mengisi tanggal.

### 3. Hasil dan Pembahasan

Berikut Merupakan Implementasi dari penerapan framework Flask dan Python yang dipakai dalam mengembangkan modul ini. Dalam menjalankan aplikasi ini, dimasukkan kode yang ada pada Kode Program 1 yang terdapat pada file *WSGI.py*, yang di isikan host dan port yang kita tentukan.

#### Kode Program 1

```
from application import create_app

app = create_app()

if __name__ == "__main__":
    app.run(host="0.0.0.0", port=9999, debug=True)
```

Konfigurasi yang menggunakan framework Flask bisa dibuat seperti Kode Program 2 yang terdapat pada file *INIT.py*, dapat dilakukan inialisasi aplikasi, mendaftarkan blueprint, dan mengelola konfigurasi.

#### Kode Program 2

```
from flask import Flask
from application.controller.ErrorController import err
from application.controller.loginController import login
from application.controller.lovController import lov_controller
from application.controller.onlineOrderController import myOrder

app = Flask(__name__)
app.register_blueprint(lov_controller)
app.register_blueprint(login)
app.register_blueprint(err)
app.register_blueprint(myOrder)
```

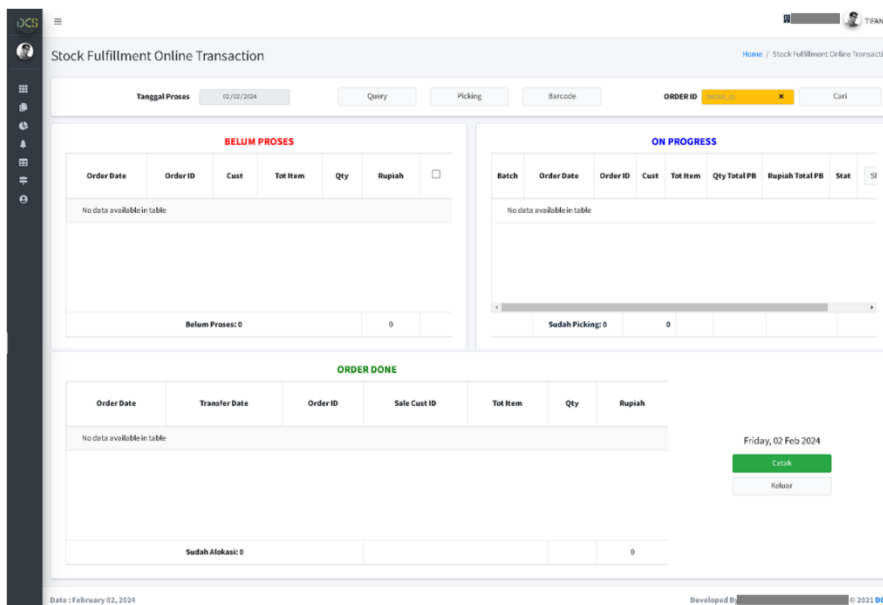
Kode Program 3 Merupakan Controller Aplikasi, dan blueprint yang didaftarkan pada `init.py`, mengatur route url yang akan dipakai, get request API untuk mengambil data dan di kembalikan dalam JSON untuk disimpan pada Frontend dan juga membawa title, dan tanggal yang telah di atur formatnya degan menggunakan fungsi dari library `datetime`.

### Kode Program 3

```
from flask_login import login_required
from flask import flash, render_template, jsonify, Blueprint, session, request, abort,
send_file
from application.decorator import role_required, stockawal_required
import requests
from config import Config
from datetime import datetime as date

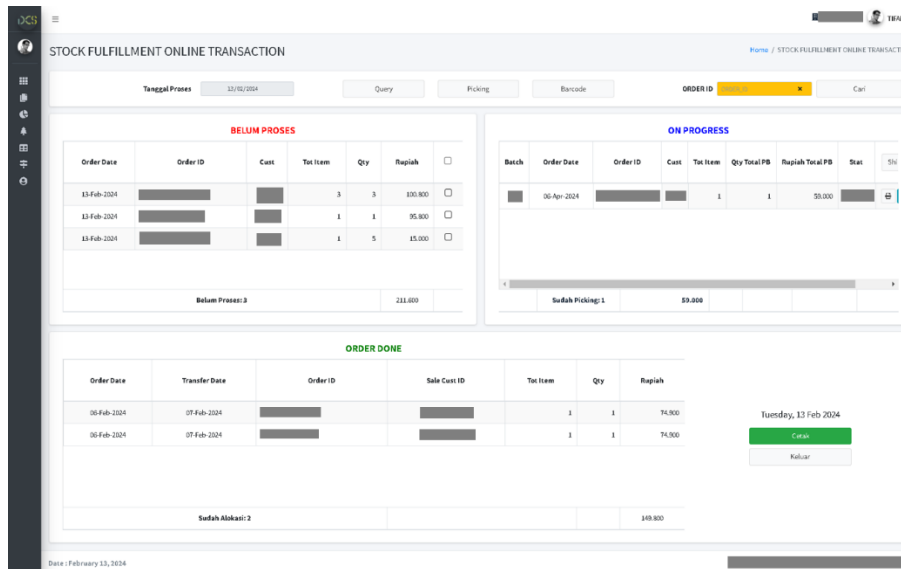
myOrder = Blueprint("myOrder", __name__, url_prefix="/onlineOrder")
URL = 'onlineOrder/'

@onlineOrder.route("/", methods=['GET'])
@login_required
@role_required('***')
def index():
    headers = {"Content-Type": "application/json", "X-Access-Token":
session["token"]}
    response = requests.get(url=Config.BACKEND_URL(cabang=session["***"]) + URL +
'header', headers=headers)
    responseJSON = response.json()
    data = {}
    if (do something)= error handler
    x = date.now()
    dt = x.strftime("%A, %d %b %Y")
    return render_template("/onlineOrder.html", title="Stock Fulfillment Online
Transaction", data=data,dt=dt)
```



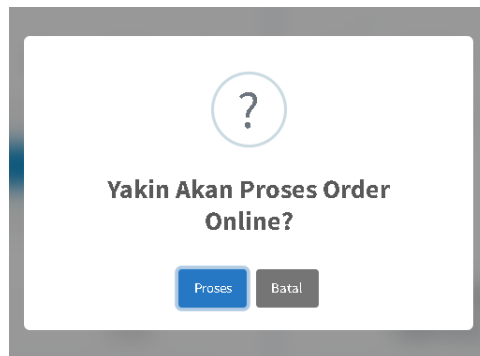
Gambar 2. Halaman Modul *Stock Fulfillment Online Transaction*

Gambar 2 adalah tampilan dari modul *Stock Fulfillment Online Transaction* yang menjadi tempat untuk melakukan pemenuhan stok barang di request dari gudang atau *warehouse* ke toko. *User* dapat melakukan pengambilan data yang di request dengan menekan tombol *Query*.



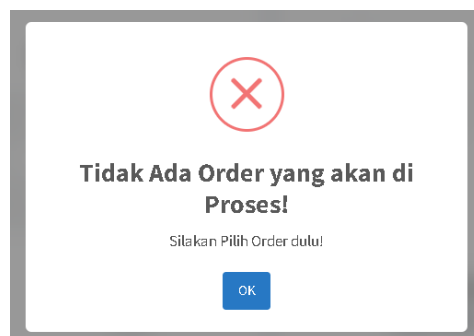
Gambar 3. Halaman Modul *Stock Fulgillment Online Transaction* setelah query

Gambar 3 adalah tampilan ketika user telah melakukan proses mengambil poses pengambilan data, setelah itu user dapat menchecklist data yang terdapat pada tabel belum proses sebelum melakukan proses picking. Setelah proses



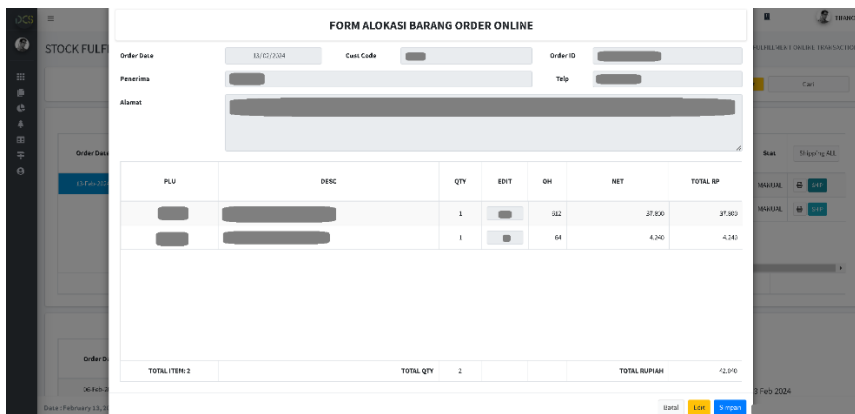
Gambar 4. Pop up konfirmasi

Gambar 4 adalah pop up konfirmasi yang akan muncul ketika ada data yang akan di picing atau diproses.



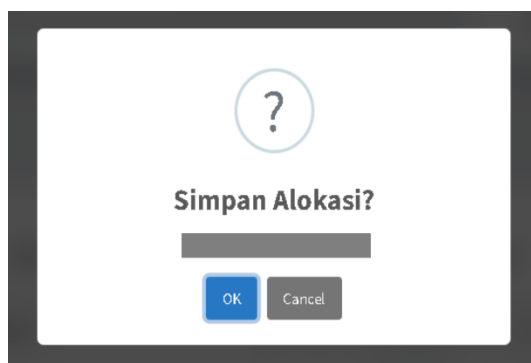
Gambar 5. Notifikasi tidak ada *Order* yang di pilih

Gambar 5 adalah *popup* yang akan muncul ketika sistem tidak mendapatkan data yang akan di *picking* atau proses, dikarenakan *user* tidak menchecklist data yang terdapat pada tabel belum *order*.



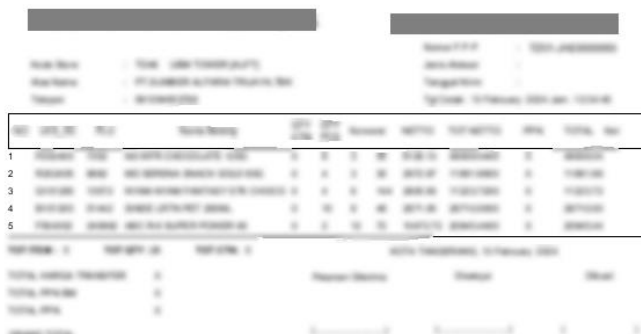
Gambar 6. Tampilan proses *Ship*

Gambar 6 adalah adalah tampilan dari proses *Ship* barang yang telah selesai *picking* atau proses. User dapat mengecek kembali alamat, penerima, dan nomor teleponnya, terdapat juga detail dari pesanan barang yang dipesan.



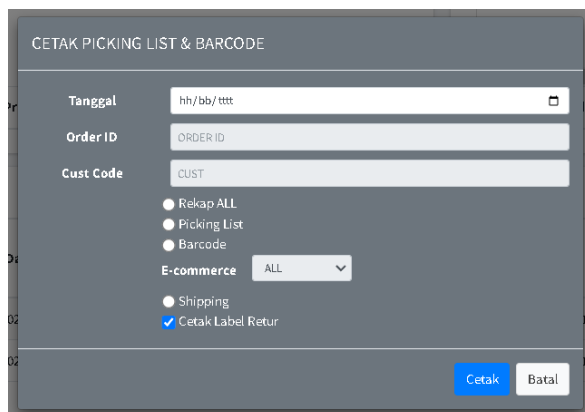
Gambar 7. Notifikasi Konfirmasi Simpan Alokasi

Gambar 7 adalah adalah tampilan *popup* yang muncul ketika *user* menekan tombol simpan yang berada pada Gambar 6. *Popup* ini juga menampilkan order id dari data atau barang yang di akan di simpan.



Gambar 8. Tampilan Cetak PDF

Gambar 8 adalah hasil dari cetak dalam bentuk pdf yang digunakan untuk proses bisnis selanjutnya.



Gambar 9. Tampilan Cetak Picking List & Barcode

Tabel 1. Hasil testing tim QA

No	Heading	Output	Status
1	ketika stok awal belum dilakukan akan mempopup “proses stok awal belum dilakukan”	Sesuai System Requirement	OK
2	Jika Data yang akan di proses tidak ada akan menampilkan “Tidak ada data yang akan di proses”	Sesuai System Requirement	OK
3	User mencari data dengan order id	Sesuai System Requirement	OK
4	Terdapat validasi ketika proses picking dan shipping	Sesuai System Requirement	OK
5	Saat picking dan shipping, data berhasil terupdate	Sesuai System Requirement	OK
6	Mencetak Barcode	Sesuai System Requirement	OK

Tabel 1 merupakan hasil pengujian yang telah dilakukan oleh tim QA (*Quality Assurance*) PT. XYZ. Proses QA dilakukan agar mengetahui bahwa aplikasi yang dibuat sudah berjalan seperti yang diharapkan pengguna dan sesuai juga dengan versi Oracle Form. Hasil dari pengembangan modul *Stock Fulfillment Online Transaction* menunjukkan bahwa penggunaan teknologi framework Flask dan Python banyak memberi manfaat dalam pengurangan biaya pemeliharaan di dibandingkan dengan teknologi seperti Oracle Form yang memiliki biaya pemeliharaan yang besar dan di nilai kurang menguntungkan. Aplikasi yang dikembangkan dengan teknologi ini dapat digunakan atau dioperasikan di banyak platform. Dari pengujian tim QA didapati bahwa modul *Stok Fulfillment Online Transaction* dapat memenuhi kriteria yang diinginkan oleh pengguna dan juga dengan versi Oracle Form. Dengan adanya pengurangan biaya pemeliharaan aplikasi, PT. XYZ dapat menggunakan biaya anggaran dalam pengembangan dan keperluan bisnis yang lain.

#### 4. Kesimpulan

Dalam penelitian dilakukan pengembangan modul *Stock Fulfillment Online Transaction* pada aplikasi DCS (*Distribution Center System*) dengan menggunakan *framework* Flask dan Python. Pengujian yang dilakukan tim QA juga dilakukan agar dapat memastikan bahwa modul dapat berjalan dengan baik dan sudah sesuai dengan kebutuhan pengguna dan juga sesuai dengan versi Oracle Form. Hasil pengembangan menunjukkan bahwa menggunakan *framework* Flask dan Python memberikan banyak manfaat seperti aplikasi dapat digunakan di banyak platform yang berbeda dan juga dapat mengurangi

biaya pemeliharaan di bandingkan dengan versi Oracle Form. Dengan adanya hal itu, PT. XYZ dapat mengalokasikan biaya anggaran perusahaannya kepada hal lainnya. Adapun beberapa saran yang bisa diterapkan untuk kedepannya seperti melakukan evaluasi lebih lanjut dan pemeliharaanm aplikasi dengan baik secara berkala baik tampilan dan juga fitur serta hal lainnya yang perlu dalam kebutuhan bisnis kedepannya.

## 5. Daftar Pustaka

- Alwi, M. Y., Parlyna, R., & Hidayat, N. (2021). Pengaruh Kualitas Layanan dan Kepuasan Pelanggan terhadap Loyalitas Pelanggan Pengguna Ojek Online X. *Jurnal Bisnis, Manajemen, Dan Keuangan*, 2(1), 292-308.
- Anjani, R. G. (2019). Peran Sistem Informasi Terhadap Operasonal Retail. *Jurnal Ekonomi Manajemen Sistem Informasi*, 1(1), 79-86.
- Bramasta, A., & Ikram, A. D. (2023). PENGARUH KETERSEDIAAN BARANG DAN KECEPATAN PENGIRIMAN TERHADAP KEPUASAN PELANGGAN. *Jurnal Kewirausahaan dan Inovasi*, 2(2).
- Budianto, A. J., & Saian, P. O. N. (2023). Pengembangan Modul Inventory Management pada Aplikasi Master Distribution Centre System Menggunakan Framework Flask di PT XYZ. *Jurnal JTik (Jurnal Teknologi Informasi dan Komunikasi)*, 7(2), 201-207. DOI: <https://doi.org/10.35870/jtik.v7i2.714>.
- Ghimire, D. (2020). Comparative study on Python web frameworks: Flask and Django.
- Irsyad, R. (2018). Penggunaan Python Web Framework Flask Untuk Pemula.
- Lapian, S. L. H. V. J. (2017). Pengaruh Strategi Competitive Advantage Terhadap Perilaku Konsumen Di Bisnis Ritel (Studi Pada Indomaret Dan Alfamart, Manado). *Jurnal LPPM Bidang EkoSosBudKum (Ekonomi, Sosial, Budaya, dan Hukum)*, 3(2), 1-13.
- Microsoft Community. (2023, November 16). Oracle Forms And Reports 6i Installation On Windows 10. *Microsoft Community*. <https://answers.microsoft.com/en-us/windows/forum/all/oracle-forms-andreports-6i-installation-on/1d47c8b8-d963-49a8-a397-423846ffb0a4>.
- Novindri, G. F., & Saian, P. O. N. (2022). Implementasi Flask Pada Sistem Penentuan Minimal Order Untuk Tiap Item Barang Di Distribution Center Pada Pt Xyz Berbasis Website. *Jurnal Mnemonic*, 5(2), 81-85. DOI: <https://doi.org/10.36040/mnemonic.v5i2.4670>.
- Octaviani, K., Komara, M. A., & Kurniawan, I. (2022). Analisis Kesuksesan Aplikasi Alfagift Menggunakan Model Delon dan Mclean Studi Kasus Alfa Express Rest Area Km 72b. *Jurnal Informatika Teknologi dan Sains (Jinteks)*, 4(3), 173-178.
- Praba, A. D., & Safitri, M. (2020). Studi perbandingan performansi antara mysql dan postgresql. *jurnal khatulistiwa informatika*, 8(2). DOI: <https://doi.org/10.31294/jki.v8i2.8851>.
- Saragih, R. R. (2016). Pemrograman dan bahasa Pemrograman. *STMIK-STIE Mikroskil*, 1-91.

Vyshnavi, V. R., & Malik, A. (2019). Efficient way of web development using python and flask. *Int. J. Recent Res. Asp*, 6(2), 16-19.

Walingkas, H. L., & Saian, P. O. N. (2023). Penerapan Framework Flask pada Pembangunan Sistem Informasi Pemasok Barang. *Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi)*, 7(2), 227-234. DOI: <https://doi.org/10.35870/jtik.v7i2.729>.