

# PENGEMBANGAN APLIKASI *REPORTING* PENENTUAN KUANTITAS MAKSIMUM PADA SISTEM APLIKASI OPERASI PERUSAHAAN XYZ MENGGUNAKAN *FLASK FRAMEWORK*

Johannes Satria Putra Kase\*<sup>1)</sup>, Pratyaksa Ocsa Nugraha Saian <sup>2)</sup>

1. Universitas Kristen Satya Wacana, Indonesia
2. Universitas Kristen Satya Wacana, Indonesia

## Article Info

**Kata Kunci:** *Flask framework*; *Pandas*; Sistem Reporting; *Waterfall*

**Keywords:** *Flask framework*; *Pandas*; *Reporting System*; *Waterfall*

## Article history:

Received 19 May 2023

Revised 2 June 2023

Accepted 16 June 2023

Available online 1 December 2023

## DOI :

<https://doi.org/10.29100/jipi.v8i4.4231>

\* Corresponding author.

Johannes Satria Putra Kase

E-mail address:

[johannes.sp.kase@gmail.com](mailto:johannes.sp.kase@gmail.com)

## ABSTRAK

Perkembangan teknologi memberikan peluang untuk meningkatkan kinerja dan efisiensi bagi para pelaku bisnis. Perusahaan XYZ, salah satu perusahaan retail terbesar di Indonesia dengan 16.490+ gerai beroperasi di Indonesia, merespons perkembangan tersebut dengan melakukan migrasi aplikasi Operation Application System (OAS) yang menggunakan Oracle Form ke Python. Teknologi Oracle sudah tidak relevan lagi dikarenakan biaya lisensi yang tinggi dan hanya dapat beroperasi pada satu platform saja. Berdasarkan latar belakang tersebut, framework Flask dan Pandas digunakan dalam penelitian ini untuk membangun sistem reporting pada aplikasi OAS berbasis web pada modul Penentuan Kuantitas Maksimum (PKM). Metode penelitian yang digunakan adalah Model Lima Tahap (Mantap) yang terdiri dari Penelitian Pendahuluan, Pengembangan Model, Validasi Model, Uji Efektivitas Model, dan Diseminasi. Waterfall digunakan sebagai metode pengembangan sistem. Hasil Pengujian dengan Unit Testing menunjukkan sudah tidak ada bug dan sudah tidak ada error pada validasi textfield TAG dan textfield tanggal. Pengujian dengan Black Box menunjukkan hasil valid sesuai dengan yang diharapkan. Berdasarkan hasil penelitian ini maka dapat dinyatakan bahwa pengembangan aplikasi ini memenuhi kebutuhan pengguna. Kelebihan dari sistem yang dibangun menggunakan Python dengan framework Flask dan Pandas ini adalah pembaharuan teknologi, dapat dioperasikan di berbagai platform, dan memangkas biaya karena tidak perlu biaya lisensi.

## ABSTRACT

Technological advancements offer opportunities to enhance performance and efficiency for business stakeholders. XYZ Company, one of the largest retail companies in Indonesia with 16,490+ stores operating nationwide, responds to these developments by migrating its Operation Application System (OAS) from Oracle Form to Python. The use of Oracle technology has become irrelevant due to high licensing costs and its limited operation on a single platform. In light of this background, the Flask and Pandas frameworks are utilized in this research to construct a web-based reporting system within the OAS application, specifically in the Maximum Quantity Determination (PKM) module. The research follows the Five-Stage Model (Mantap), which includes Preliminary Research, Model Development, Model Validation, Model Effectiveness Testing, and Dissemination. The Waterfall methodology is employed for system development. Unit Testing demonstrates the absence of bugs or errors in validating the TAG textfield and date textfield. Black Box testing yields expected and valid results. Based on these findings, it can be concluded that the application development fulfills user requirements. The advantages of utilizing Python with the Flask and Pandas frameworks include technology updates, multi-platform operability, and cost reduction by eliminating licensing fees.

## I. PENDAHULUAN

**T**EKNOLOGI informasi secara terus menerus dikembangkan agar dapat terus meningkatkan keefektifan dan kemanfaatannya. Dalam menjalankan perusahaan dibutuhkan teknologi yang mampu menangkap atau mengambil, menyimpan, mengelola, dan mengembalikan informasi dengan cepat, tepat, dan efisien. Perusahaan membutuhkan sistem informasi yang dapat menghasilkan informasi yang dibutuhkan agar dapat mencapai tujuan tertentu [1]. Perusahaan melakukan perubahan dari sistem konvensional yang masih menggunakan kertas, menjadi sistem terkomputerisasi yang dapat memudahkan penggunaannya. Berbagai sektor melakukan banyak perubahan ini, salah satunya adalah sektor penjualan retail [2].

Salah satu perusahaan retail terbesar di Indonesia adalah PT. XYZ yang telah berdiri lebih dari 20 tahun dengan 16.490+ gerai beroperasi di Indonesia, dan 1.200+ toko yang beroperasi di Filipina. PT. XYZ aktif beroperasi dengan jumlah transaksi sekitar 5 juta transaksi setiap harinya [3]. Oleh karena jumlah transaksi yang tinggi tersebut dibutuhkan sistem informasi yang dapat membantu dalam proses bisnis barang dari gudang dan toko. Proses bisnis yang menjadi fokus dari penelitian ini adalah pengelolaan persediaan per toko dengan dilakukan Penentuan Kuantitas Maksimum (PKM), yaitu kondisi di mana penetapan penentuan suatu barang di suatu toko untuk setiap item atau barangnya.

Dalam menjalankan proses bisnis, PT. XYZ memerlukan sistem informasi yang dapat mencatat laporan agar dapat membantu dalam proses bisnis penentuan kuantitas maksimum setiap barang pada setiap toko. PT XYZ membangun modul Laporan PKM untuk mencatat dan mencetak laporan menggunakan Oracle Form, yang merupakan bagian dari aplikasi *Operation Application System* (OAS).

Namun teknologi yang digunakan saat ini untuk mencatat dan mencetak laporan, dirasa sudah tidak cocok untuk dipakai. Teknologi Oracle Form yang digunakan dalam pembuatan sistem informasi di PT. XYZ [4], sudah waktunya digantikan dengan teknologi yang lebih efektif dan tepat guna. Teknologi Oracle memiliki biaya lisensi yang tinggi, aplikasi yang rumit, membutuhkan pelatihan khusus untuk penggunaannya, dan membutuhkan *hardware* dengan spesifikasi tinggi, hanya dapat beroperasi pada satu platform saja [5]. Teknologi yang lebih tepat guna menjadi kebutuhan yang penting bagi sistem informasi PT. XYZ. Python dengan *framework* Flask dan Pandas menjadi jawaban dari kebutuhan ini

Python bersifat *open-source* yang memungkinkan pengembangan menghemat biaya bebas biaya lisensi [6] Bahasa pemrograman Python menjadi salah satu Bahasa pemrograman tingkat tinggi yang sering digunakan karena memiliki karakteristik sintaksis yang mudah dimengerti [7] Python dapat digunakan untuk berbagai jenis pengembangan perangkat lunak karena Python hadir dengan *library* bawaan yang dapat diperluas sehingga menjadi modular dan *reuseable* [8].

*Framework* Flask dipilih sebagai kerangka kerja pembangunan aplikasi ini karena Flask tergolong dalam *micro-framework*. Flask mengurangi penggunaan *resource memory* dan memiliki fitur-fitur yang memberikan fleksibilitas kepada penggunaannya [9]. Selain itu, Flask menyediakan *library* yang bertujuan mempercepat pembangunan aplikasi tanpa harus membuatnya dari nol [10]. Flask Framework dikatakan fleksibel karena dapat dikembangkan di atas kerangka inti untuk menambahkan fitur aplikasi seolah-olah bagian dari Flask itu sendiri [11]. Flask Framework juga mendukung ekstensi tambahan yang digunakan pada aplikasi yang sedang dikembangkan seperti ekstensi untuk *database*, validasi pada formulir dan *upload handling* [12].

Pandas berasal dari kata Panel Data. P-data adalah data multidimensi yang mencakup pengukuran dari waktu ke waktu. Wes McKinney menciptakan Panda pada tahun 2015. Struktur data dan fungsi-fungsi bahasa pemrograman Python sepenuhnya disediakan oleh Pandas, sebuah pustaka perangkat lunak, untuk pemrosesan data terstruktur yang lebih cepat dan lebih sederhana. Dengan potensi manipulasi data yang dapat diadaptasi, mengelola lembar kerja dan *database* relasional menjadi mudah. Selain itu, Panda memprioritaskan fungsi pengindeksan progresif [13].

Berdasarkan latar belakang tersebut, penelitian ini dilakukan yaitu pengembangan sistem menggunakan Python dengan Flask framework dan Pandas pada modul Laporan PKM dalam membangun sistem *reporting* pada aplikasi OAS berbasis web. Penelitian-penelitian rancang bangun sistem informasi *reporting* sudah dilakukan oleh peneliti-peneliti sebelumnya. Penelitian-penelitian terkait dapat memberikan kontribusi berupa ide dan gagasan penelitian.

Sebuah penelitian dilakukan oleh Herdiansah dkk bertujuan untuk mengubah pengerjaan laporan yang masih konvensional menjadi laporan menggunakan sistem informasi berbasis komputer. Hasil penelitian berupa sistem informasi *monitoring* dan *reporting quality control* proses laminating berbasis web dengan menggunakan *framework* Laravel yang dapat membantu proses pencatatan dan pembuatan laporan *Quality Control*. Perancangan sistem informasi ini menggunakan *Unified Modeling Language* (UML) dan dalam tahap pengembangan sistem

digunakan bahasa pemrograman PHP dengan *framework* Laravel. Kemudian sistem yang dihasilkan diuji dengan metode pengujian Black Box. Sistem yang dibangun telah berhasil membantu *monitoring* dan *reporting* secara efisien dan akurat, di samping itu sistem ini juga menghasilkan laporan dalam bentuk *dashboard* (diagram) yang memudahkan manajemen dalam menganalisa laporan hasil produksi. Sistem ini didasarkan pada web sehingga tidak memerlukan pengaturan khusus pada komputer pengguna, cukup menggunakan komputer yang terhubung ke jaringan dan server serta menggunakan peramban untuk dapat mengakses sistem yang telah dikembangkan. Dengan demikian terkonfirmasi bahwa pengerjaan laporan yang masih konvensional menjadi laporan menggunakan sistem informasi berbasis komputer sudah menjadi tuntutan jaman [14]. Penelitian yang membangun sistem informasi *monitoring* dan *reporting* dengan sistem berbasis web dengan menggunakan *framework* Laravel ini memberikan gagasan membangun sistem informasi serupa namun dengan mengimplementasikan *framework* Flask.

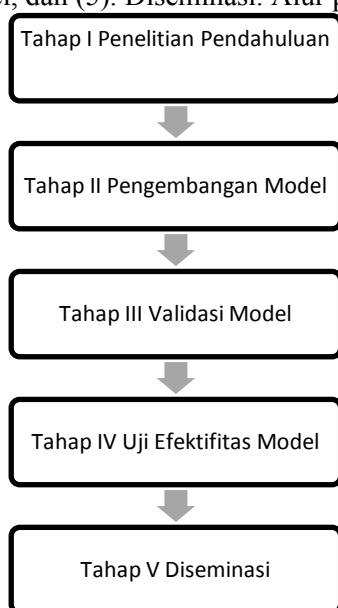
Penelitian yang dilakukan oleh Reiss dkk berfokus pada penggunaan Pandas Library dari Python. Pengelompokan *kueri*, desain grafis data dalam format tabel dapat dilakukan dengan berbagai dukungan yang ada di *library* ini. Pandas Library menawarkan struktur data dan operasi untuk memanipulasi tabel numerik dan deret waktu. Dalam penelitian ini dilakukan pengumpulan data struktur yang dibuat secara manual diplot, dianalisis, dan divisualisasikan dengan Python menggunakan berbagai Pandas Library, *file* kumpulan data struktur mengakses berbagai format seperti xls, csv, pdf dan banyak lagi. Penelitian ini diimplementasikan pada *database* karyawan yang dibuat secara acak untuk melakukan berbagai operasi dan visualisasi data dengan Python menggunakan library Pandas [15]. Penelitian tersebut memberikan kontribusi gagasan menggunakan Pandas Library untuk mengembangkan menu *reporting* pada aplikasi OAS.

Penelitian yang dilakukan oleh Snehkunj dkk mendiskusikan bahwa Text Extensions for Pandas memungkinkan untuk membangun aplikasi *Neuro Linguistic Programming* (NLP) *end-to-end* sambil merepresentasikan semua data internal aplikasi dengan DataFrames. Text Extensions for Pandas ini menyediakan kumpulan ekstensi tipe data, operasi khusus NLP, dan integrasi perpustakaan NLP yang mengubah Pandas DataFrames menjadi struktur data universal untuk mengelola data mesin yang mengalir melalui aplikasi NLP. *Text Extensions* ini tersedia secara bebas sebagai paket Python yang dapat dipasang dan sebagai kode sumber [16]. Berbeda dengan penelitian tersebut yang menggunakan NLP, pada penelitian ini akan menggunakan XlsxWriter.

Dengan demikian tiga penelitian tersebut turut berkontribusi memberikan informasi dan gagasan dalam penelitian ini. Akan tetapi penelitian ini berbeda dengan tiga penelitian tersebut yaitu bahwa penelitian ini membangun sistem *reporting* pada aplikasi OAS dengan menggunakan Flask Framework dan Library Pandas.

## II. METODE PENELITIAN

Penelitian ini dilakukan dalam 5 tahapan, yaitu: (1). Penelitian Pendahuluan, (2). Pengembangan Model, (3). Validasi Model, (4). Uji Efektivitas Model, dan (5). Diseminasi. Alur penelitian dideskripsikan pada Gambar 1.

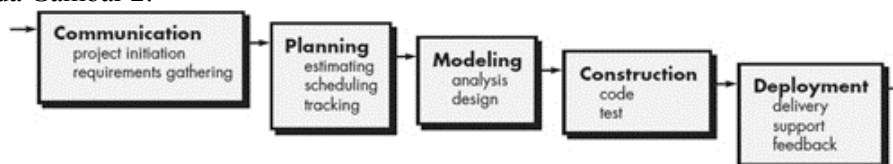


Gambar. 1. Prosedur Penelitian dan Pengembangan

Gambar 1 menjelaskan tahapan penelitian secara berurutan. Tahapan penelitian ini adalah Model Lima Tahap (Mantap) [17] hasil modifikasi dari tahapan Penelitian dan Pengembangan Level 4 rancangan Sugiyono [18].

Metode Pengembangan Sistem yang digunakan dalam penelitian ini adalah *Waterfall*. Metode pengembangan *Waterfall* adalah merupakan pendekatan *System Development Life Cycle* (SDLC) paling awal yang digunakan untuk pengembangan perangkat lunak. Model *Waterfall* pertama kali diperkenalkan oleh Winston Royce sekitar tahun 1970, sehingga dianggap sebagai model klasik. Namun demikian model ini mempunyai keunggulan bersifat sistematis dan berurutan sehingga paling banyak dipakai dalam *Software Engineering* (SE). Tahap demi tahap yang dilalui dilakukan secara sistematis, harus menunggu selesainya tahap sebelumnya, dan berjalan berurutan [19].

Lima tahap metode *Waterfall* akan dilakukan secara sistematis dan berurutan. Tahapan metode *Waterfall* dideskripsikan pada Gambar 2.



Gambar. 2. Metode Waterfall (Pressman)

Tahapan awal model *Waterfall* menurut referensi Pressman adalah: *Communication* (Project Initiation & Requirements Gathering). Komunikasi dengan klien sangat perlu dilakukan supaya tujuan dapat tercapai. Dalam hal ini team developer melakukan komunikasi dengan user mengenai permasalahan yang ada. Hasil komunikasi diketahui bahwa PT XYZ sedang dalam proses migrasi aplikasi Operation Application System (OAS) yang menggunakan Oracle Form ke Python. Dikarenakan teknologi Oracle sudah tidak relevan lagi dikarenakan biaya lisensi yang tinggi dan membutuhkan tenaga kerja yang tampil menggunakan Oracle Form. Selain dari hasil komunikasi, juga dilakukan kajian pustaka dari jurnal, buku, dan sumber-sumber lain sehingga didapatkan gagasan untuk membangun sistem reporting pada OAS ini menggunakan Python dengan *framework* Flask dan Pandas

Tahapan berikutnya adalah *Planning* (Estimating, Scheduling, Tracking). Pada tahap ini dilakukan perencanaan mengenai perkiraan tugas teknis yang akan dilakukan, potensi risiko, jadwal kerja, dan melakukan pelacakan proses pengerjaan sistem. Team developer mulai membagi pekerjaan kepada setiap talen dengan mempertimbangkan kemampuan, risiko, dan jadwal kerja kepada setiap talennya.

Kemudian tahap *Modeling* (*Analysis & Design*). Di tahapan ini dilakukan desain dan pembuatan model arsitektur sistem yang menekankan pada desain struktur data, arsitektur perangkat lunak, tampilan antarmuka, dan algoritma program. Tujuan dari tahap ini adalah untuk memperoleh gambaran keseluruhan tentang apa yang akan dilakukan. Pada tahapan ini dilakukan perancangan/desain UML modul Penentuan Kuantitas Maksimum (PKM) pada Oracle Form sebagai landasan dan batas-batas pengembangan aplikasi. Desain UML menggunakan *Use Case*, *Activity Diagram*, dan *Class Diagram* seperti yang ditunjukkan oleh Gambar 3, Gambar 4, dan Gambar 5

Selanjutnya tahap *Construction* (*Code & Test*). Pada tahapan *Construction* ini dilakukan *Code* dan *Test* yang merupakan proses pembangunan aplikasi menggunakan Bahasa pemrograman Python dan menggunakan Flask sebagai *framework*. Setelah kode program selesai, dilakukan pengujian (*testing*) terhadap aplikasi menggunakan metode Unit Testing dan Black Box. Pengujian dilakukan untuk menemukan kesalahan yang mungkin terjadi untuk kemudian diperbaiki.

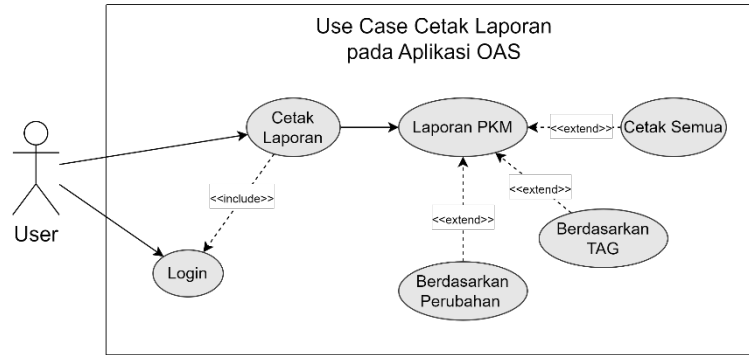
Tahap akhir adalah *Deployment* (*Delivery, Support, Feedback*) Tahapan *Deployment* merupakan tahapan implementasi aplikasi, pemeliharaan aplikasi secara berkala, perbaikan aplikasi, evaluasi aplikasi, dan pengembangan aplikasi berdasarkan umpan balik yang diberikan agar sistem dapat tetap berjalan dan berkembang sesuai dengan fungsinya. Hal ini dilakukan agar kualitas produk yang di keluarkan dapat terjaga kualitasnya.

### III. HASIL DAN PEMBAHASAN

Hasil dari penelitian ini dapat dilihat pada tiga bagian tahapan yaitu pada Perancangan Sistem, Pengembangan/Implementasi, dan Pengujian.

Berikut adalah perancangan sistem bagian UML untuk menggambar kan perancangan sistem dalam modul Penentuan Kuantitas Maksimum (PKM) dengan menggunakan *Use Case Diagram*, *Class Diagram*, dan *Activity Diagram*

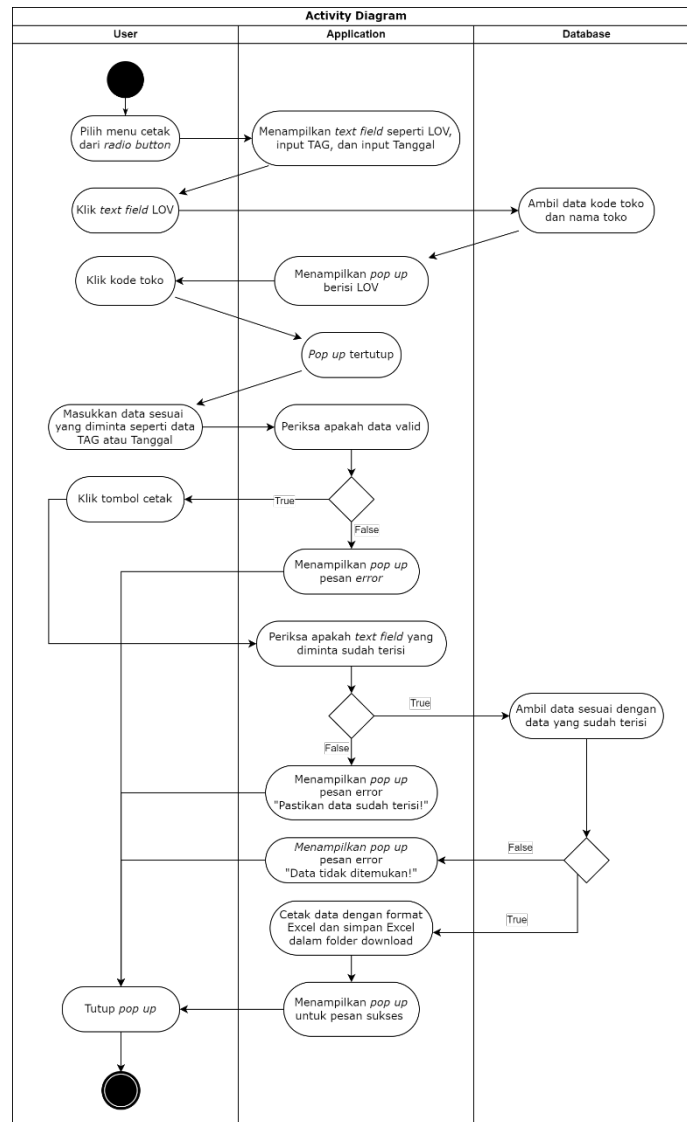
### A. Use Case Diagram



Gambar. 3. Use Case Diagram

Gambar 3 adalah *Use Case Diagram* yang digunakan pada penelitian ini. *Use Case Diagram* ini digunakan untuk menggambar hubungan sistem pada program dengan pengguna program. Aktor dalam diagram ini digambarkan dalam bentuk *stickman*. *User* sebagai Aktor merupakan karyawan yang akan menjalankan aplikasi ini. Sistem program aplikasi ini digambarkan dalam bentuk persegi yang berisikan *use case* yang digambarkan dengan bentuk oval. *User* akan diberikan tiga pilihan menu cetak yaitu Cetak Semua (Print All PKM), Cetak Berdasarkan TAG (Print by TAG), dan Cetak Berdasarkan Perubahan (Print by Perubahan PKM). Semua menu cetak terletak di dalam Laporan PKM yang di mana *User* perlu melakukan *Login* terlebih dahulu agar dapat masuk ke dalam Laporan PKM.

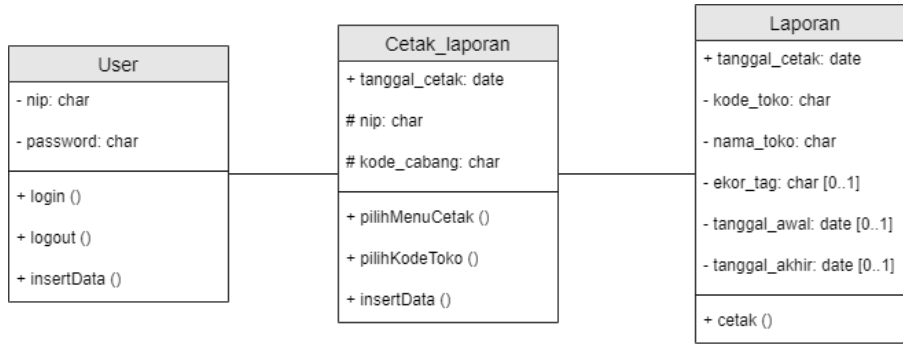
B. Activity Diagram



Gambar. 4. Activity Diagram

Gambar 4 merupakan gambar *Activity Diagram* dari modul PKM. User perlu memilih menu dengan cara klik salah satu *radio button* yang sudah disediakan. Setelah itu aplikasi akan menampilkan *text field* LOV dan *text field* sesuai dengan menu yang dipilih. User dapat memilih kode toko dengan cara klik pada *text field* LOV yang akan menampilkan *pop up* berisikan *list* dari kode toko yang tersedia, dan kode toko dapat dipilih dengan cara klik salah satu kode toko yang tersedia. User dapat mengisi data selanjutnya seperti mengisi TAG jika memilih untuk cetak berdasarkan TAG, atau memasukkan rentang tanggal yang akan dicetak jika memilih untuk mencetak berdasarkan rentang tanggal yang diinginkan. User dapat mencetak data PKM dengan cara klik tombol cetak. Aplikasi akan memeriksa apakah seluruh data yang diminta sudah terisi atau belum, setelah itu akan mengambil data dari *database* berdasarkan data yang sudah dimasukkan.

### C. Class Diagram

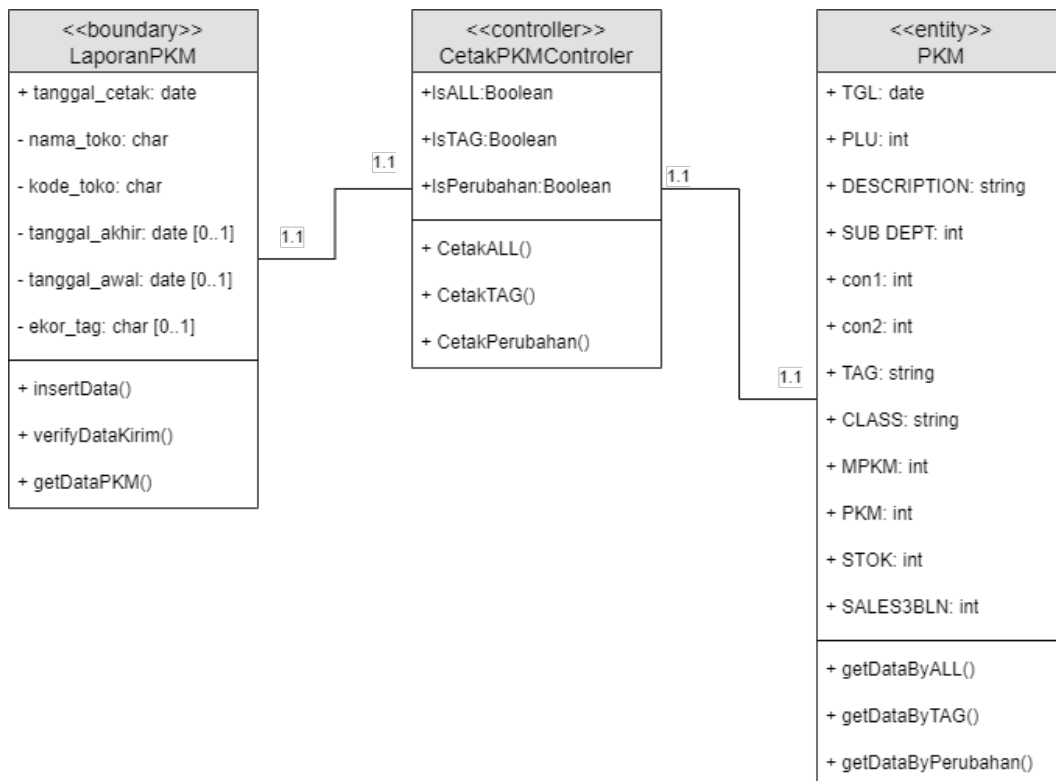


Gambar. 5. Class Diagram

Gambar 5 adalah *Class Diagram* yang digunakan pada penelitian ini. *Class Diagram* digunakan untuk menggambarkan dengan skema dari program penelitian ini. *Class Diagram* ini dibagi menjadi tiga komponen yaitu komponen atas, komponen tengah, dan komponen bawah.

Komponen atas berisikan nama *class*, terdapat dalam tiga *class* pada diagram yaitu; User, Cetak Laporan, dan Laporan. Komponen tengah berisikan atribut dari *class*. Atribut pada *class* User ada dua yaitu nomor induk pegawai (NIP) dan *password*. *Class* Cetak Laporan terdapat tiga atribut yaitu tanggal cetak, nip, dan kode cabang. *Class* Laporan terdapat enam atribut yaitu tanggal cetak, kode toko, nama toko, ekor *tag*, tanggal awal, dan yang terakhir tanggal akhir. Komponen bawah berisikan operasi yang dilakukan seperti pada *class* User terdapat operasi *login*, *logout*, dan *insert* data. *Class* Cetak Laporan terdapat operasi pilih menu cetak, pilih kode toko, dan *insert* data. *Class* Laporan terdapat operasi cetak yang berfungsi untuk mencetak laporan.

Dalam desain *Class Diagram*, terdapat dua bagian yang perlu dipahami. Gambar 5 adalah *Class Diagram* yang menjelaskan model domain, yang merupakan gambaran umum dari basis data. Bagian kedua adalah bagian dari pola program *Model View Controller* (MVC), yang terdiri dari *boundary class* sebagai antarmuka, *control class* sebagai tempat penempatan algoritma, dan *entity class* sebagai tabel dalam basis data serta *kueri* program, seperti tampak pada Gambar 6.



Gambar. 1. Class Diagram MVC

Pada Gambar 6 menunjukkan *boundary class* dengan nama LaporanPKM di mana *user* dapat memasukkan data untuk dikirimkan ke *controller*, fungsi *verifyDataKirim* untuk memverifikasi data yang akan dikirim, dan

getDataPKM untuk mengambil data kode toko yang akan digunakan dalam LOV. CetakPKMController sebagai *controller class* dengan fungsi CetakALL, CetakTAG, dan CetakPerubahan untuk mencetak laporan. *Entity class* yang berisikan fungsi-fungsi untuk mengambil data yang digunakan untuk mencetak laporan seperti getDataByALL, getDataByTAG, dan getDataByPerubahan.

#### D. Implementasi

Seperti yang sudah dibahas sebelumnya, penelitian ini mengimplementasikan Flask sebagai kerangka dalam membangun aplikasi OAS. Berikut adalah implementasi Flask Framework pada kode Python. Dalam menjalankan aplikasi yang menggunakan Flask Framework, perlu menentukan *host* dan *port* sesuai dengan kebutuhannya seperti Kode Program 1.

TABEL 1  
KODE PROGRAM UNTUK MENJALANKAN APLIKASI DENGAN FLASK

```
from flask import Flask
app = Flask(__name__)
# ...
if __name__ == '__main__':
    app.run(debug=True)
```

Tabel 1 adalah *source code* dalam menjalankan aplikasi web menggunakan Flask. Perintah pada baris pertama bertujuan untuk *import* kelas Flask dari modul Flask. Baris kedua berisikan *object* dari kelas Flask. Baris terakhir *source code*, merupakan konfigurasi dari aplikasi yang dijalankan pada *localhost* dengan *port* 5000.

Selanjutnya, dibutuhkan kode pemrograman untuk mendaftarkan *blueprint* dan *route*. Untuk mendaftarkan *blueprint* dan *route*, penelitian ini menggunakan API. Berikut *source code* untuk mendaftarkan *blueprint* dan *route*.

TABEL 2  
KODE PROGRAM UNTUK MENDAFTARKAN BLUEPRINT

```
from flask import render_template, redirect, url_for, flash, g

from connect import app
from flask_restful import Api
from LaporanPkm_controller import *

app.config['SECRET_KEY'] = os.urandom(24)
api = Api(app)
# ...

api.add_resource(CetakLaporanPKM, '/cetak/')

# ...

@app.route('/report_pkm', methods=['GET', 'POST'])
def report_pkm():
    if not g.user:
        flash('Sesi berakhir, coba masuk lagi.')
        return redirect(url_for('user_home'))
    else:
        return render_template('report_pkm.html')
```

Tabel 2 berisikan tentang cara mendaftarkan *blueprint* dan *route*. Agar Flask dapat menjalankan berbagai *Uniform Resource Locator (URL)*, *blueprint* dapat didaftarkan terletak pada baris **api.add\_resource** di mana parameternya berisi *class* yang sudah dibuat di *controller* dan *endpoint* yang harus diisi jika ingin menggunakan dua URL yang berisi *class* yang sama. Dalam melakukan *mapping* antar URL, dapat dicapai dengan menggunakan **@app.route** yang dapat diberikan situasi *if else* seperti kode pemrograman Tabel 2.

Tabel 3 adalah *controller* yang digunakan pada modul penelitian ini. Kegunaan dari Flask pada kode program ini untuk mengambil data dari *database*. Berikut *source code* dari *controller* untuk membuat fungsi cetak.



TABEL 3  
KODE PROGRAM UNTUK FUNGSI CETAK

```
import os
import pandas as pd

from flask import request, jsonify
from flask_restful import Resource
from connect import ConnectBranch
from globals.db import DatabaseServer

class CetakLaporan(Resource):
    def __init__(self):
        self.connect_branch = ConnectBranch()
        self.db = DatabaseServer

    def get(self):
        try:
            #..
            # Set file name
            filename = r'Laporan_PKM_All.xlsx'

            # Change into data frame
            df = pd.DataFrame.from_dict(data)

            # Set writer for dataframe
            excel = pd.ExcelWriter(filename, engine='xlsxwriter')

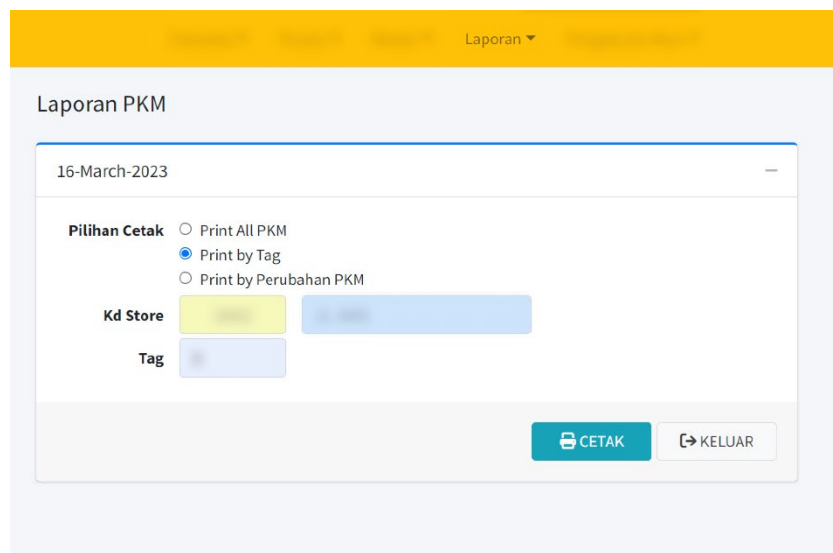
            # Shift the dataframe down one row in the Excel file.
            df.to_excel(excel, sheet_name='Sheet1', startrow=0, index=False)

            excel.save()

            self.result["data"] = dataQueryry
            self.result["status"] = True

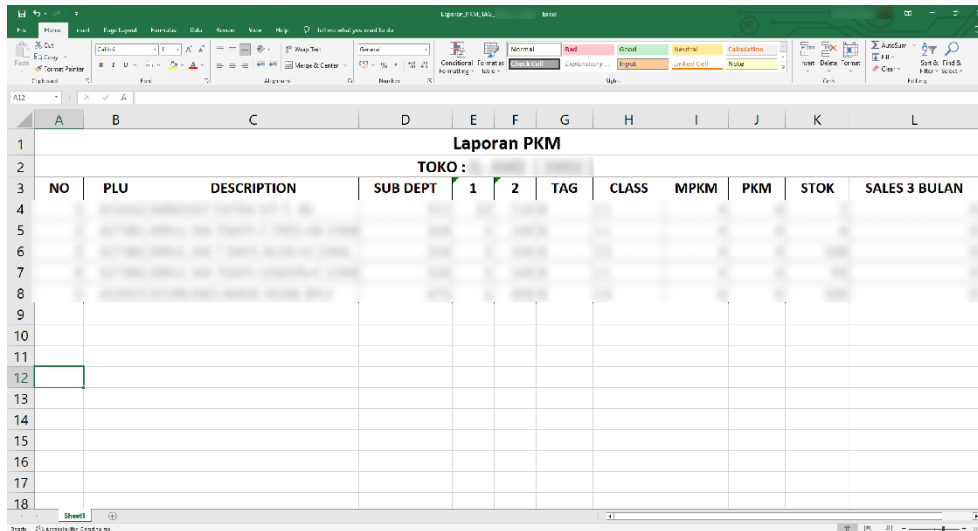
        except Exception as e:
            import traceback
            traceback.print_exc()
            error_message = str(e)
            self.result["message"] = error_message
            return self.result
```

Berikut adalah tampilan halaman utama yang digunakan pada penelitian ini.



Gambar. 7. Halaman Laporan PKM

Gambar 7 merupakan tampilan dari halaman untuk mencetak laporan PKM. User dapat memilih satu dari tiga menu cetak PKM yang terdiri dari “Print All PKM”, “Print by Perubahan PKM”, dan yang sedang ditunjukkan pada Gambar 7 yaitu “Print by Tag”. Setelah memilih menu, aplikasi akan memunculkan *textfield* sesuai dengan menu yang dipilih. Dalam kasus ini, *textfield* yang ditampilkan yaitu untuk memasukkan *Tag* yang akan diberikan oleh *User*. Setelah terisi semua, *user* dapat mulai mencetak dengan cara klik pada tombol “CETAK” yang sudah tersedia.



Gambar. 8. Laporan PKM

Gambar 8 merupakan tampilan dari Laporan PKM yang dicetak oleh *user*. Laporan yang dicetak akan berubah tergantung dengan pilihan menu dan kode toko yang dimasukkan oleh *user*. Data yang sudah disimpan dalam bentuk *dataframe* dapat dicetak dan menghasilkan laporan dalam format Excel seperti pada **Gambar 8**.

Pengujian sistem yang dilakukan pada penelitian ini dilakukan dengan dua macam pengujian. Pengujian menggunakan metode Unit Testing dan Black Box. Pengujian yang pertama menggunakan metode Unit Testing. Metode ini dilakukan dengan menguji setiap unit program secara terpisah untuk memastikan setiap unit tersebut berfungsi dengan baik. Tujuan dari unit testing adalah untuk mengidentifikasi *bug* dan memperbaiki sebanyak mungkin agar layak untuk digunakan oleh *user*.

Dalam mencapai tujuan penelitian ini, dilakukan pengujian fungsi-fungsi yang terdapat dalam modul *reporting* pada aplikasi *Operation Application Sistem* (OAS). Fungsi yang diperiksa yaitu fungsi validasi dari *textfield* TAG, hal ini dilakukan agar saat *user* memasukkan ekor TAG sistem dapat memastikan apakah ekor TAG terdaftar dalam sistem atau tidak. Fungsi validasi juga terletak pada *textfield* tanggal, hal ini untuk memastikan tanggal kedua lebih besar dari tanggal pertama. Karena pada dasarnya jika memberikan batasan tanggal selalu diawali tanggal pertama lebih kecil dari tanggal kedua. Kedua fungsi validasi di periksa dengan memberi *input* yang benar dan yang salah, saat memasukkan *input* yang tidak sesuai maka akan memunculkan pesan *error*.

Setelah semua data yang diminta terisi, fungsi yang diperiksa selanjutnya yaitu fungsi cetak. Fungsi ini memiliki beberapa fungsi di dalamnya sebagai berikut; fungsi yang berperan untuk memeriksa apakah data yang di isikan terdapat di *database*, selanjutnya fungsi mengambil data dari *database* dan mengolah data ke dalam bentuk Excel. Jika data yang dimasukkan oleh *user* tidak ditemukan maka akan menampilkan pesan *error* bahwa data tidak ditemukan.

Pengujian pertama dengan Unit Testing menunjukkan hasil sudah tidak ada *bug*. Pengujian pada fungsi validasi dari *textfield* TAG, *textfield* tanggal, dan fungsi cetak menunjukkan hasil sudah tidak ada *error*.

Pengujian yang kedua dilakukan oleh bagian *quality assurance* (QA) dari proyek ini. Pengujian ini dilakukan oleh satu orang QA yang ditugaskan untuk melakukan testing pada aplikasi ini. Pengujian yang dilakukan menggunakan metode Black Box, berikut hasil dari pengujian yang sudah dilakukan. Pengujian ini dilakukan selama satu hari.

TABEL 4  
 TABEL BLACK BOX TESTING

No	Skenario Pengujian	Hasil yang Diharapkan	Hasil Pengujian	Status Pengujian
1	Memasukkan atau memilih LOV kode toko dari database.	Dapat memasukkan atau memilih LOV kode toko dari database	Sesuai yang diharapkan	Valid
2	Memilih antara menu cetak <i>Print All PKM, Print by TAG, Print by Perubahan PKM</i> menggunakan <i>radio button</i> magnetic induction	Dapat memilih menu menggunakan <i>radio button</i> yang berisi menu cetak <i>Print All PKM, Print by TAG, Print by Perubahan PKM</i>	Sesuai yang diharapkan	Valid
3	<i>Text field</i> muncul berdasarkan menu cetak yang dipilih.	Dapat memunculkan <i>text field</i> berdasarkan menu cetak yang dipilih	Sesuai yang diharapkan	Valid
4	Menampilkan pesan <i>error</i> saat ekor TAG yang di isi tidak terdaftar	Dapat menampilkan pesan <i>error</i> saat ekor TAG yang di isi tidak terdaftar	Sesuai yang diharapkan	Valid
5	Menampilkan pesan <i>error</i> saat tanggal ke-dua lebih kecil dari pada tanggal pertama	Dapat menampilkan pesan <i>error</i> saat tanggal ke-dua lebih kecil dari pada tanggal pertama	Sesuai yang diharapkan	Valid
6	Menampilkan pesan <i>error</i> jika data tidak ada atau di temukan	Dapat menampilkan pesan <i>error</i> jika data tidak ada atau di temukan	Sesuai yang diharapkan	Valid
7	Mengambil data dari database untuk di cetak dengan format Excel	Dapat mengambil data dari database untuk di cetak dengan format Excel.	Sesuai yang diharapkan	Valid

Berdasarkan hasil dari kedua pengujian sistem dapat disimpulkan bahwa pengembangan aplikasi OAS menggunakan Flask *framework* dan Pandas ini dapat memenuhi kebutuhan *user*. Hasil penelitian membuktikan bahwa sistem *reporting* yang dibangun sesuai digunakan untuk kegiatan *monitoring* pada aplikasi OAS. Selain itu juga terdapat keuntungan dari penggunaan Python dengan Flask *framework* dan Pandas yaitu pemangkasan biaya lisensi, pembaharuan teknologi, dan dapat dioperasikan di berbagai platform. Penelitian ini menunjukkan hasil yang menambah value dari penelitian terdahulu yang dilakukan oleh Herdiansah dkk yaitu dengan diimplementasikannya *framework* Flask.

#### IV. KESIMPULAN

Berdasarkan hasil penelitian dapat disimpulkan bahwa pengembangan aplikasi OAS pada modul Laporan PKM menggunakan Python dengan *framework* Flask dan Pandas berhasil membangun sistem *reporting* yang dapat memenuhi kebutuhan *user*. *Output* dari sistem yang dikembangkan ini berupa Excel, sehingga dapat diberikan saran untuk pengembangan aplikasi selanjutnya adalah menghasilkan *output* berbeda seperti PDF atau Word. Selain itu penggunaan bahasa Python dapat lebih diperluas dengan menggunakan validasi-validasi yang disediakan oleh Python, dan juga penggunaan *error handling* dapat lebih dikhususkan agar lebih mudah terbaca.

#### DAFTAR PUSTAKA

- [1] H. Sulistiani, Setiawansyah, and V. H. Saputra, "Penerapan Codeigniter Dalam Pengembangan Sistem Pembelajaran Dalam Jaringan," *J. CoreIT*, vol. 6, no. 2, pp. 89–95, 2020.
- [2] Liliana, D. T. Absari, and F. B. Benaly, "Pengembangan Aplikasi Retail Dengan Sistem Konfigurasi," *Teknika*, vol. 9, no. 2, pp. 156–162, 2020, doi: 10.34148/teknika.v9i2.305.
- [3] PT Sumber Alfaria Trijaya Tbk, "Laporan Tahunan 2021 PT Sumber Alfaria Trijaya Tbk," [alfamart.co.id](http://alfamart.co.id), 2021.
- [4] D. A. B. Prasetyo and Y. A. Susetyo, "Implementasi Information Schema Database Pada Postgre SQL Untuk Pembuatan Tabel Informasi Dengan Menggunakan Python Di PT XYZ," *J. Tek. Informatika dan Sist. Inf.*, vol. 9, no. 3, pp. 1961–1972, 2022.
- [5] Fiona *et al.*, "Analisis Prediksi Pemilihan Sistem Erp Pada Pt Gudang Garam Tbk Berdasarkan Proses Rantai Pasok Yang Digunakan," *Transekonomika Akuntansi, Bisnis dan Keuang.*, vol. 2, no. 3, pp. 95–112, 2022, doi: 10.55047/transekonomika.v2i3.132.
- [6] N. C. Laksmi, A. Setyanto, and F. W. Wibowo, "Kajian Pustaka Terstruktur Mengenai Analisis Audio Menggunakan Filterisasi Dengan Python," *J. Inf. J. Penelit. dan Pengabd. Masy.*, vol. 6, no. 2, pp. 16–22, 2020, doi: 10.46808/informa.v6i2.178.
- [7] Python Software Foundation, "What is Python? Executive Summary," Python.org, 2019.
- [8] K. Adawadkar, "Python Programming-Applications and Future," *Int. J. Adv. Eng. Res. Dev.*, no. April, pp. 1–4, 2017.
- [9] C. Wijayanto and Y. A. Susetyo, "Implementasi Flask Framework Pada Pembangunan Aplikasi Sistem Informasi Helpdesk (SIH)," *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.)*, vol. 07, no. 03, pp. 858–868, 2022, [Online]. Available: <https://jurnal.stkipppgritulungagung.ac.id/index.php/jipi/article/view/3161/1328>
- [10] D. Ghimire, "Comparative study on Python web frameworks: Flask and Django," *Metropolia University of Applied Sciences*, 2020. [Online]. Available: <http://www.theseus.fi/handle/10024/339796>
- [11] D. F. Ningtyas and N. Setiyawati, "Implementasi Flask Framework pada Pembangunan Aplikasi Purchasing Approval Request Flask Framework Implementation in Development Purchasing Approval Request Application," *J. Janitra Inform. dan Sist. Inf.*, vol. 1, no. 1, pp. 19–34, 2021, doi: 10.25008/janitra.v1i1.120.

- [12] Pallets, "Documentation Flask," 2022. <https://flask.palletsprojects.com/en/2.0.x/> (accessed Nov. 19, 2022).
- [13] K. Ashwitha, K. Rachmana, L. Y. Reddy, S. Khan, and S. Sundari, "Graphical User Interface for Shuffling of Sections based on Student Profile," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 9, no. VI, pp. 2752–2757, 2021, doi: 10.22214/ijraset.2021.35552.
- [14] A. F. Harismawan, A. P. Kharisma, and T. Afirianto, "Analisis Perbandingan Performa Web Service Menggunakan Bahasa Pemrograman Python , Php ,," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 1, pp. 237–245, 2017, [Online]. Available: <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/781>
- [15] F. Reiss, B. Cutler, and Z. Eichenberger, "Natural Language Processing with Pandas DataFrames," *Proc. 20th Python Sci. Conf.*, no. Scipy, pp. 49–57, 2021, doi: 10.25080/majora-1b6fd038-006.
- [16] R. Snehkunj, K. Vachiyatwala, and C. Author, "Data Analysis Using Pandas Library of Python," *Acta Sci. Comput. Sci.*, vol. 4, no. 3, pp. 37–41, 2022.
- [17] S. Sumarni, "Model Penelitian Dan Pengembangan (R&D) Lima Tahap (MANTAP)," Yogyakarta, 2019.
- [18] Sugiyono, *Metode Penelitian Kuantitatif, Kualitatif, dan R&D*. Bandung: CV Alfabeta, 2016.
- [19] R. Pressman, *Rekayasa Perangkat Lunak: Pendekatan Praktisi*. Yogyakarta: Andi, 2015.