



Pengembangan Modul *Inventory Management* pada Aplikasi *Master Distribution Centre System* Menggunakan *Framework* Flask di PT XYZ

Andre Josiah Budianto ^{1*}, Pratyaksa Ocsa Nugraha Saian ²

^{1,2} Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Kota Salatiga, Provinsi Jawa Tengah, Indonesia.

article info

Article history:

Received 15 July 2022

Received in revised form

24 November 2022

Accepted 15 February 2023

Available online April 2023

DOI:

<https://doi.org/10.35870/jti.k.v7i2.714>

Keywords:

Inventory Management; Flask;
Python; Retail.

Kata Kunci:

Inventory Management; Flask;
Python; Retail.

abstract

PT XYZ is a retail business with over 15,000 outlets. Along with the development of information technology, every retail business must adapt to existing technological developments, including PT XYZ which requires new technology. PT XYZ already has inventory management system, but the system continues to increase operating costs every year. Therefore, PT XYZ needs to replace the old inventory management system with a web-based system using the Flask framework to reduce operating costs that continue to grow. This research was conducted in 4 stages, namely analysis stage, design stage, coding stage and testing stage. The result of this research is inventory management module built using the Flask framework and information related to developing inventory management module. Based on the research conducted, an inventory management module was produced where this module successfully met the business needs as evidenced by the results of the Black Box tests and User Acceptance Tests performed.

abstrak

PT XYZ merupakan perusahaan yang bergerak pada bidang usaha bisnis retail dan memiliki 15.000 lebih gerai toko. Seiring dengan berkembangnya teknologi informasi setiap perusahaan bisnis retail perlu menyesuaikan dengan perkembangan teknologi yang ada, termasuk PT XYZ yang memerlukan adanya teknologi yang baru. PT XYZ sudah memiliki sistem inventory management tetapi sistem tersebut terus mengalami kenaikan biaya operasional setiap tahunnya. Oleh karena itu, PT XYZ memerlukan adanya pengganti dari sistem inventory management sebelumnya menjadi sistem berbasis web dengan menggunakan framework Flask guna menekan biaya operasional yang terus bertambah. Penelitian ini dibuat melalui 4 tahapan yaitu tahap analisis, tahap desain, tahap pengkodean, dan tahap pengujian. Hasil dari penelitian ini adalah modul inventory management yang dibangun menggunakan framework Flask dan informasi terkait pengembangan modul inventory management. Berdasarkan penelitian yang dilakukan dihasilkan modul inventory management dimana modul ini berhasil memenuhi kebutuhan perusahaan dibuktikan dengan hasil Black Box testing dan user acceptance test yang dilakukan.

Author. Email: 672018008@student.uksw.edu ^{1}, pratyaksa.ocs@uksw.edu ².

1. Latar Belakang

XYZ merupakan perusahaan yang bergerak pada bidang usaha bisnis *retail*. PT XYZ memiliki 15.000 lebih gerai toko yang tersebar di seluruh Indonesia. Perusahaan ini menjalankan bisnisnya dengan menjual barang kebutuhan rumah tangga sehari-hari langsung kepada konsumen melalui toko-toko yang dimiliki. Bisnis *retail* merupakan aktivitas bisnis yang melibatkan penjualan barang dan jasa secara langsung kepada konsumen akhir [1]. Seiring dengan berkembangnya teknologi informasi, setiap perusahaan bisnis *retail* perlu menyesuaikan dengan perkembangan teknologi yang ada [2]. Dalam persaingan pada bidang usaha bisnis *retail*, perusahaan-perusahaan sudah mulai mengembangkan usahanya dalam bidang teknologi. Tidak hanya untuk bertahan dalam persaingan antar perusahaan, namun guna mempermudah perusahaan dalam pengelolaan data secara efektif dan efisien. Penguasaan teknologi informasi akan membantu perusahaan dalam efisiensi penyediaan barang, efektifitas proses bisnis dan keakuratan informasi permintaan pasar [3].

Pada PT XYZ ditemukan hal serupa dimana PT XYZ memerlukan adanya teknologi yang baru salah satunya dalam bentuk aplikasi berbasis web yang dibangun menggunakan bahasa Python untuk mendukung jalannya perusahaan. PT XYZ memerlukan adanya sistem terbaru untuk manajemen stok yang ada di setiap toko yang dimiliki agar stok yang ada pada setiap toko dapat terkontrol dan juga mempermudah pengelolaan stok setiap toko. PT XYZ sudah memiliki sistem *inventory management* tetapi sistem yang dimiliki oleh perusahaan terus mengalami kenaikan biaya operasional setiap tahunnya. *Inventory* atau persediaan adalah stok bahan yang digunakan untuk memudahkan produksi atau untuk memuaskan permintaan pelanggan [4] sedangkan *Inventory management* atau manajemen inventori adalah mengelola atau mengatur agar barang-barang yang dikelola agar jumlahnya tidak berlebihan sehingga bisa menekan biaya serendah mungkin, tetapi jumlahnya juga harus selalu cukup untuk memastikan permintaan pelanggan tetap terpenuhi dengan baik [5].

Oleh karena itu, PT XYZ memerlukan adanya pengganti dari sistem yang dibangun menggunakan

Oracle Forms menjadi sistem berbasis *web* dengan menggunakan *framework* Flask guna menekan biaya operasional yang terus bertambah. Flask adalah sebuah *web framework* yang ditulis dengan bahasa Python dan tergolong sebagai jenis *microframework*. Flask berfungsi sebagai kerangka kerja aplikasi dan tampilan dari suatu web. Pengembang dapat membuat web yang terstruktur dan dapat mengatur *behaviour* suatu web dengan lebih mudah dengan menggunakan Flask dan bahasa Python [6], Python merupakan salah satu bahasa pemrograman tingkat tinggi (*High Level Language*). Python dirancang untuk memberikan kemudahan bagi *programmer* melalui segi efisiensi waktu, kemudahan dalam pengembangan dan kompatibilitas dengan sistem [7].

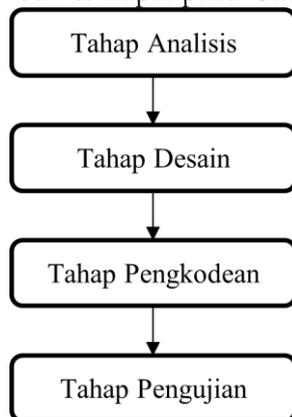
Berdasarkan permasalahan yang ada, dilakukan Pengembangan Modul *Inventory Management* pada Aplikasi *Master Distribution Centre System* menggunakan *Framework* Flask di PT XYZ. Adapun tujuan dari penelitian ini karena perlu adanya pembaharuan sistem yang sudah ada. *Framework* Flask dipilih karena *framework* ini merupakan salah satu *framework* yang paling populer saat ini dan *framework* ini dapat meningkatkan kemampuan aplikasi yang kompleks [8]. Adapun referensi dalam melakukan penelitian ini adalah pada penelitian yang berjudul “Pengembangan Modul *Inventory Management* pada Sistem Informasi *Point of Sale* berbasis Web menggunakan Metode *Incremental*” oleh Aziz Yanuar Saputra membahas tentang modul *inventory management* dibutuhkan dalam membantu pencatatan, pengawasan, serta pengecekan stok secara *realtime* yang dapat mempermudah pegawai dalam melakukan pekerjaannya. Pada penelitian ini menyebutkan bahwa *inventory management* yang dikemas dalam sistem informasi sangatlah bermanfaat untuk mencapai ketepatan dan kecepatan proses bisnis [9]. Lalu Penelitian yang berjudul “Penerapan Teknologi Enterprise Resource Planning Menggunakan Modul *Inventory Management* dengan Metode FIFO Berbasis Website Pada CV Pandan Mas Sampit” oleh Nur Septia Dwi Hapsari, dkk, disimpulkan bahwa sistem informasi yang dibuat untuk proses stok barang dapat meminimalisir adanya selisih stok barang dan juga sistem ini dapat membantu memperlancar bisnis proses dari perusahaan tersebut [10].

Penelitian yang berjudul “Model Pengembangan Sistem Informasi Akademik Berbasis *User Centered Design* Menerapkan *Framework* Flask Python” oleh

Ronaldo Kristoforus Ngantung dan Ineke Pakereng menyimpulkan bahwa manfaat dari penggunaan *framework* Flask adalah membuat pembangunan aplikasi menjadi lebih ringan karena Flask termasuk dalam *microframework* yang tidak memerlukan *library* dalam *framework* dan Flask merupakan *framework* yang mudah digunakan dan dipahami karena fiturnya sederhana namun dapat melakukan pembangunan aplikasi berbasis web yang cukup rumit [11]. Lalu, pada penelitian yang berjudul “Implementasi API Master Store Menggunakan Flask, REST, dan ORM di PT XYZ” oleh Brian Pratama Putra dan Yerymia Alfa Susetyo menggunakan *framework* Flask karena Flask merupakan *microframework* dan berbeda dari kebanyakan *framework* karena Flask jauh lebih ringan dan cepat serta mudah untuk diimplementasikan sehingga dengan penggunaan *framework* Flask, pengembang atau *programmer* dapat membuat sebuah website yang ringan dan terstruktur [12]. Keterkaitan penelitian ini dengan penelitian-penelitian sebelumnya adalah modul *inventory management* dibutuhkan dalam membantu para pemilik usaha untuk memanajemen stok dan meminimalisir kesalahan dalam manajemen stok, kemudian *framework* Flask dapat mempermudah pengembangan *website* karena *framework* ini mudah untuk dipahami dan fiturnya yang sederhana namun dapat melakukan pembangunan aplikasi basis web yang cukup rumit. Penelitian terdahulu menguatkan alasan kenapa perlu adanya pengembangan lebih lanjut dari modul *inventory management* yang sudah ada dan juga penggunaan *framework* Flask sebagai *framework* yang efisien dalam pembangunan modul ini.

2. Metode Penelitian

Tahapan penelitian terdapat pada Gambar 1 [13].



Gambar 1. Tahapan Penelitian

Tahapan penelitian pada Gambar 1 dijelaskan sebagai berikut:

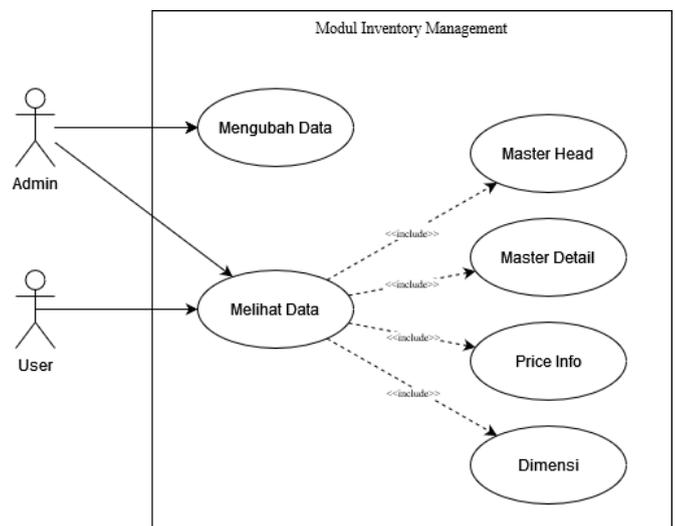
- 1) Tahap analisis masalah dilakukan untuk menganalisis masalah yang ada dari modul terdahulu dan menentukan teknologi serta metode yang akan digunakan pada penelitian.
- 2) Tahap selanjutnya adalah tahap desain, pada tahap ini dilakukan perancangan *Use Case Diagram* dan perancangan *Activity Diagram*.
- 3) Tahap pengkodean adalah tahap pembuatan modul berdasarkan rancangan yang sudah dibuat menggunakan *framework* Flask.
- 4) Tahap pengujian adalah tahap pengujian modul yang telah dibuat dengan menggunakan *Black Box Testing* dan *User Acceptance Test*. Dimana *User Acceptance Test* akan dilakukan oleh 20 calon pengguna aplikasi dan pengukuran hasil pengujian menggunakan skala Likert.

3. Hasil dan Pembahasan

Hasil dari penelitian ini adalah modul *Inventory Management* pada aplikasi *Master Distribution Center System* menggunakan *framework* Flask. Terdapat beberapa tahap utama dalam pembahasan ini yaitu tahap desain, tahap pengkodean, dan tahap pengujian.

1) Tahap Desain

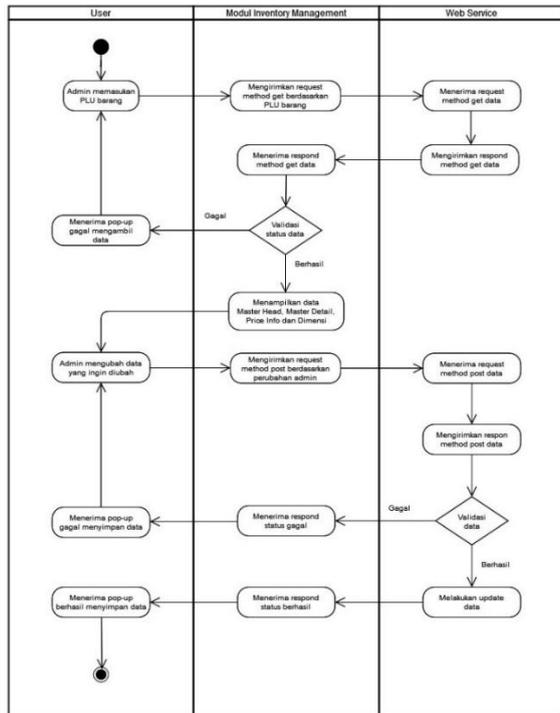
Berikut merupakan hasil dari desain modul :



Gambar 1. Use Case Diagram Modul Inventory Management

Gambar 2 merupakan Use Case Diagram dari modul *inventory management*. Pada diagram tersebut *admin*

dapat melakukan dua operasi yaitu mengubah dan melihat data. Pada operasi melihat data, *user* dan *admin* dapat melihat data dari barang yang dipilih. Data tersebut meliputi data *master head*, *master detail*, *price info*, dan dimensi. Sebelum menjalankan operasi mengubah data, *admin* harus melakukan operasi melihat data terlebih dahulu. Lalu pada operasi mengubah data, *admin* dapat mengubah tipe lokasi, lokasi, dan stok barang sementara.



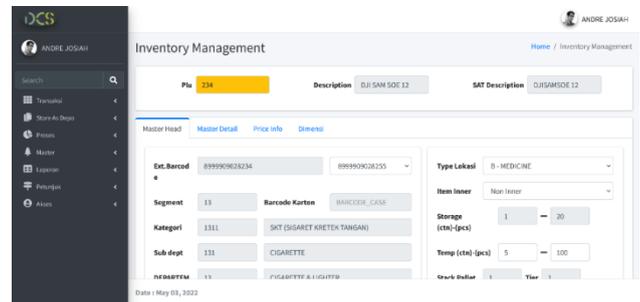
Gambar 2. Activity Diagram Modul Inventory Management

Gambar 3 merupakan Activity Diagram dari modul *inventory management*. Alur pada Gambar 3 merupakan alur mengubah data pada modul *inventory management*. Activity Diagram yang terjadi adalah *admin* memasukkan PLU barang untuk mendapatkan data dari barang yang dicari, lalu dilakukan validasi oleh modul terhadap *respond* yang diberikan oleh *web service*. PLU merupakan pengenal dari setiap barang yang ada atau bisa disebut ID dari suatu barang. Apabila *respons* yang diberikan adalah berhasil maka *admin* akan mendapatkan data, data yang didapat meliputi data *master head*, *master detail*, *price info*, dan dimensi. Jika gagal maka *admin* akan mendapatkan *pop-up* bertuliskan data gagal ditampilkan. Setelah data berhasil ditampilkan *admin* dapat mengubah data sesuai dengan yang diperlukan. Apabila data yang ingin diubah *admin* tidak sesuai maka *web service* akan

mengirimkan status gagal ke modul *inventory management* maka *admin* akan mendapat *pop-up* bertuliskan gagal menyimpan data. Sebaliknya apabila status berhasil maka, data akan disimpan dan *admin* akan mendapatkan *pop-up* bertuliskan data berhasil disimpan.

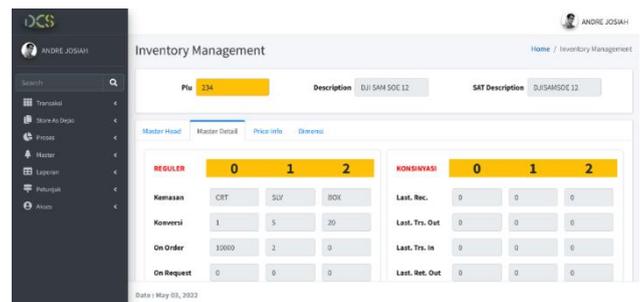
2) Tahap Pengkodean

Dari hasil desain modul yang telah dibuat, pembuatan modul *inventory management* dibuat menggunakan *framework* Flask. Berikut merupakan hasil dari pembuatan modul *inventory management*:



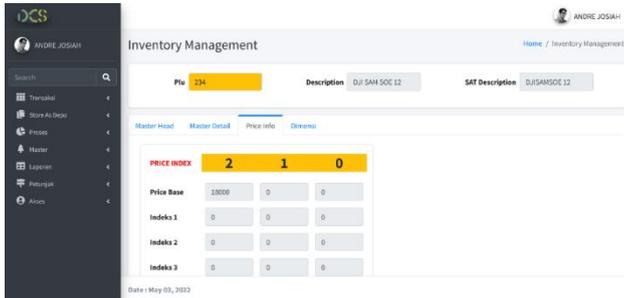
Gambar 4. Tampilan Awal Modul Inventory Management

Gambar 4 merupakan tampilan awal dari modul *inventory management*. Pada tampilan ini *user* dan *admin* dapat melihat data dari barang yang dicari hanya dengan memasukkan PLU dari barang tersebut. Jika data yang dimasukkan sesuai dengan yang ada di *database* maka data akan ditampilkan, jika gagal akan muncul *pop-up* peringatan data gagal ditampilkan. Tampilan ini akan menampilkan data yang menjadi acuan *admin* dalam pemesanan barang kepada vendor agar tidak terjadi *overstock* dan menampilkan lokasi barang tersebut akan disimpan. Pada tampilan ini juga *admin* dapat melakukan perubahan pada lokasi akan disimpannya barang dan memperbaharui stok barang sementara.



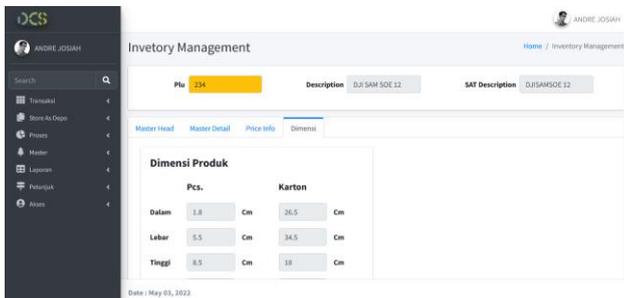
Gambar 5. Tampilan Master Detail pada Modul Inventory Management

Gambar 5 merupakan tampilan *master detail*. Pada tampilan ini *user* dan *admin* dapat melihat data yang lebih spesifik mengenai barang yang dicari. Data yang ada pada master detail nantinya akan menjadi acuan *admin* dalam mengubah data dan juga acuan dalam mengelola stok barang yang ada agar efektif dalam pendistribusiannya ke toko cabang.



Gambar 6. Tampilan *Price Info* pada Modul *Inventory Management*

Gambar 6 merupakan tampilan *price info*. Tampilan ini berfungsi untuk menampilkan data terkait harga dasar dari barang dan harga alternatif untuk barang yang dicari. Tampilan ini menjadi acuan untuk penetapan harga jual produk di toko cabang yang ada.



Gambar 7. Tampilan Dimensi pada Modul *Inventory Management*

Gambar 7 merupakan tampilan dimensi. Pada tampilan ini *user* dan *admin* mendapatkan informasi terkait dimensi produk secara satuan maupun karton. Tampilan ini menjadi acuan bagaimana cara mengatur tata letak penyimpanan barang dan tata letak barang pada kendaraan pengangkut agar efektif.

Kode Program 1. Fungsi *get data* pada *framework* Flask

```

1 @inventory_management.route("/get_data",
2 methods=['GET'])
3 def get_data():
4     data = request.get_json()
5     kd_store = data['kd_store']
6     plu = data['plu']
7     try:
8         result = mstDisplay.getData(plu, kd_store)

```

```

9         return jsonify(result)
10
11     except Exception as e:
12         return jsonify({'status':False,
13 'msg':str(e)})

```

Kode Program 1 merupakan fungsi pemanggilan data pada modul *inventory management*. Baris 3 merupakan proses penerimaan data dari modul berupa data JSON, setelah itu pada baris 4 dan 5 dilakukan pengambilan data berdasarkan *key* yang dibutuhkan. Kemudian pada baris ke 7 data yang diambil dilempar ke fungsi *getData()* nantinya hasil kembalian berupa data JSON. Pada baris 10 merupakan pengecualian apabila terjadi *error* pada saat pengambilan data dan akan mengembalikan status *false* serta pesan *error*.

Kode Program 2. Fungsi *update data* pada *framework* Flask

```

1 @inventory_management.route("/update",
2 methods=['POST'])
3 def update():
4     data = request.get_json()
5     try:
6         result = mstDisplay.update(data)
7         return jsonify(result)
8
9     except Exception as e:
10        return jsonify({'status':False,
11 'msg':str(e)})

```

Kode Program 2 berfungsi untuk *update data* pada modul *inventory management*. Baris 3 merupakan proses penerimaan data dari modul berupa data JSON. Pada baris 5 data yang sudah diterima dari modul dilempar ke fungsi *update()*. Apabila data berhasil diubah atau terdapat kegagalan dalam prosesnya maka akan ada kembalian dalam bentuk JSON ke modul *inventory management*.

3) Tahap Pengujian

Pada tahap pengujian dilakukan pengujian terhadap fungsional dari modul ini. Pengujian dilakukan dengan metode *Black Box Testing* dan *User Acceptance Test*. Berikut hasil dari pengujian modul:

Tabel 1. Pengujian pada modul *inventory management* dengan metode *Black Box Testing*

No	Fungsi	Skenario	Output yang diharapkan	Output	Hasil
1	Melihat Data	Plu barang tidak ada pada database	Muncul <i>pop-up</i> gagal	Muncul <i>pop-up</i> "gagal memuat data"	Valid
2	Melihat Data	Plu barang ada pada database	Menampilkan seluruh data	Seluruh data ditampilkan	Valid

3	Mengubah Data	Form diisi tidak sesuai alur	Muncul <i>pop-up</i> gagal	<i>pop-up</i> "gagal mengubah data"	Valid
4	Mengubah Data	Form diisi dengan benar	Muncul <i>pop-up</i> sukses	Muncul <i>pop-up</i> "berhasil mengubah data"	Valid

Black Box Testing adalah jenis pengujian yang berhubungan dengan perangkat lunak dengan kinerja internal yang tidak diketahui [14]. Berdasarkan pengujian menggunakan metode *Black Box Testing* yang telah dilakukan pada modul *inventory management* dapat disimpulkan bahwa modul ini sudah berjalan dengan baik secara fungsional dilihat dari status valid yang didapatkan pada setiap skenario pengujian yang dilakukan.

Setelah dilakukan tes fungsional aplikasi dilakukan juga *user testing* yang bertujuan untuk menilai apakah modul yang sudah dibuat sudah mencapai ekspektasi dan kebutuhan dari *user*. Tes ini dibuat dengan melibatkan 20 calon pengguna modul *inventory management*. Berikut merupakan hasil dari *User Acceptance Test* yang diberikan kepada *user* melalui kuesioner:

Tabel 2. Hasil *User Acceptance Test*

No	Pertanyaan	Jawaban					Jumlah	Indeks
		SS	S	C	TS	STS		
1	Apakah modul <i>Inventory Management</i> sudah sama seperti sistem yang sebelumnya?	18	2	-	-	-	98	98%
2	Apakah dengan adanya modul yang diperbaharui mempermudah <i>user</i> dalam bekerja?	15	5	-	-	-	95	95%
3	Apakah modul yang diperbaharui berfungsi lebih baik daripada sistem sebelumnya?	17	3	-	-	-	97	97%
4	Apakah dengan adanya modul ini <i>user</i> dapat bekerja secara efektif dan efisien?	14	6	-	-	-	94	94%
Total		64	16				384	96%

Berdasarkan hasil kuesioner pada Tabel 2, selanjutnya diberikan bobot penilaian pada setiap jawaban yang didapatkan. Pada kolom jawaban SS merupakan sangat setuju dengan bobot nilai 5, S merupakan setuju dengan bobot nilai 4, C merupakan cukup dengan bobot nilai 3, TS merupakan tidak setuju dengan bobot nilai 2, dan STS merupakan sangat tidak setuju dengan bobot nilai 1. Kolom jumlah merupakan total perhitungan nilai jawaban. Kolom indeks merupakan hasil persentase dari jumlah/total *user* dikali dengan 100. Hasil persentase setiap pertanyaan diukur dengan menggunakan skala Likert. Berdasarkan skala Likert kriteria interpretasi intervalnya adalah sebagai berikut : 0%-19.99% = Sangat Tidak Setuju, 20%-39% = Tidak Setuju, 40%-59.99% = Cukup, 60%-79.99% = Setuju, dan 80%-100% = Sangat Setuju [15].

Berdasarkan hasil yang di dapat pada Tabel 2, pada pertanyaan pertama indeks yang didapat adalah 98%, lalu pada pertanyaan kedua 95%, pada pertanyaan ketiga 97%, pada pertanyaan keempat 94% dan total indeks 96%. Maka berdasarkan hasil *user acceptance test* yang dilakukan pada modul *inventory management*, modul ini mendapatkan total indeks sebesar 96% yang berarti *user* sangat setuju. Maka dapat disimpulkan bahwa modul ini dapat membantu *user* dalam mengerjakan pekerjaannya secara efektif dan efisien serta membantu perusahaan dengan adanya modul yang diperbaharui.

4. Kesimpulan dan Saran

Berdasarkan penelitian yang telah dilakukan dihasilkan modul *inventory management* guna menunjang jalannya perusahaan. Modul yang dihasilkan telah memenuhi kebutuhan perusahaan dan berjalan dengan baik dilihat dari hasil pengujian yang telah dilakukan. Penelitian ini juga menghasilkan informasi terkait dengan pengembangan modul aplikasi menggunakan *framework* Flask. Hasil dari penelitian ini juga sesuai dengan hasil penelitian-penelitian terdahulu. Saran untuk pengembangan modul ini adalah dengan membuat tampilan yang lebih menarik untuk modul ini agar *user* dapat mengoperasikan aplikasi dengan lebih nyaman dan terdapat pengembangan fitur lebih lanjut sesuai dengan kebutuhan *user* guna meningkatkan kinerja dari *user* juga membantu perusahaan.

5. Daftar Pustaka

- [1] Sejati, V.A., 2015. Strategi Periklanan pada Bisnis Retail. *Jurnal ASPIKOM*, 2(4), pp.235-242. DOI: 10.24329/aspikom.v2i4.74.
- [2] Anjani, R.G., 2019. Peran Sistem Informasi Terhadap Operasional Retail. *Jurnal Ekonomi Manajemen Sistem Informasi*, 1(1), pp.79-86. DOI: 10.31933/jemsi.v1i1.49.
- [3] Budiherwanto, I., 2020. Sustainability retail di era persaingan global. *Dimensi*, 2(1), pp.34-42.
- [4] Rohmat Hidayat, W. W. & F. R. U. 2018. Pembangunan Sistem Informasi Inventory Management Pada CV. Sumber Tani - Cirebon, *UNIVERSITAS AMIKOM Yogyakarta*, 5(2302), pp. 10–73.
- [5] Widiyanto, D., 2022. Perancangan Sistem Informasi Manajemen Inventori Berbasis Web (Studi Kasus: SMK YPT Purworejo). *Jurnal Ekonomi dan Teknik Informatika*, 10(1), pp.24-31. DOI: <http://dx.doi.org/10.37601/jneti.v10i1.183>.
- [6] Irsyad, R., 2018. Penggunaan Python Web Framework Flask Untuk Pemula. Available at: [10.31219/osf.io/t7u5r](https://osf.io/t7u5r).
- [7] Saragih, R.R., 2016. Pemrograman dan bahasa Pemrograman. *STMIK-STIE Mikroskil*, pp.1-91.
- [8] Jeovano, J., 2020. 2D Data Visualization Tools Menggunakan Flask dan AngularJS. *INSYST: Journal of Intelligent System and Computation*, 2(2), pp.91-97. DOI: 10.52985/insyst.v2i2.184.
- [9] Saputra, A.Y., 2018. Pengembangan Modul Inventory Management pada Sistem Informasi Point of Sale berbasis Web menggunakan Metode Incremental. *Repository.Unugha.Ac.Id*. Available at: <http://repository.unugha.ac.id/id/eprint/711/contents>.
- [10] Hapsari, N.S.D., Mustaqiem, M. and Minarni, M., 2021. Penerapan Teknologi Enterprise Resource Planning Menggunakan Modul Inventory Management dengan Metode FIFO Berbasis Website Pada CV Pandan Mas Sampit. *TIN: Terapan Informatika Nusantara*, 2(6), pp.365-372.
- [11] Ngantung, R.K. and Pakereng, M.I., 2021. Model Pengembangan Sistem Informasi Akademik Berbasis User Centered Design Menerapkan Framework Flask Python. *Jurnal Media Informatika Budidarma*, 5(3), pp.1052-1062. DOI: 10.30865/mib.v5i3.3054.
- [12] Putra, B.P. and Susetyo, Y.A., 2020. Implementasi Api Master Store Menggunakan Flask, Rest Dan Orm Di Pt Xyz. *SISTEMASI*, 9(3), pp.543-556. DOI: 10.32520/stmsi.v9i3.899.
- [13] Soraya, A. and Wahyudi, A.D., 2021. Rancang Bangun Aplikasi Penjualan Dimsum Berbasis Web (Studi Kasus: Kedai Dimsum Soraya). *Jurnal Teknologi dan Sistem Informasi*, 2(4), pp.43-48. Available at: <http://jim.teknokrat.ac.id/index.php/JTISI>.
- [14] Salamah, U. and Khasanah, F.N., 2017. Pengujian Sistem Informasi Penjualan Undangan Pernikahan Online Berbasis Web Menggunakan Black Box Testing. *Information Management For Educators And Professionals: Journal Of Information Management*, 2(1), pp.35-46.
- [15] Fatulloh, A. and Hendro, H., 2021. Aplikasi Asesmen Lembaga Sertifikasi Profesi Polibatam Berbasis Website. *G-Tech: Jurnal Teknologi Terapan*, 5(1), pp.353-366. DOI: 10.33379/gtech.v5i1.939.