International Journal Software Engineering and Computer Science (IJSECS)
4 (1), 2024, 367-376
Published Online April 2024 in IJSECS (http://www.journal.lembagakita.org/index.php/ijsecs)
P-ISSN : 2776-4869, E-ISSN : 2776-3242. DOI: https://doi.org/10.35870/ijsecs.v4i1.2339.

RESEARCH ARTICLE Open Access

Development of a Web-Based Trading Term Application Using Flask Framework at PT. XYZ

Sarah Larasati
Faculty of Information Technology, Universitas Kristen Satya Wacana, Salatiga City, Central Java Province, Indonesia.
Email: 6720202174@student.uksw.edu

Yeremia A. Susetyo *
Faculty of Information Technology, Universitas Kristen Satya Wacana, Salatiga City, Central Java Province, Indonesia.
Corresponding Email: yeremia.alfa@uksw.edu

Received: March 8, 2024; Accepted: April 10, 2024; Published: April 30, 2024.

Abstract: The use of information technology in data management and collaboration with suppliers has increased efficiency for retail companies such as PT. XYZ. However, some companies still still need help managing data and cooperation agreement documents, which hinders the preparation of reports and has the potential for conflict. This research aims to build a web-based Trading Term application using the Flask framework at PT. XYZ. This application was developed in Python because of its simplicity and diversity of libraries. The waterfall method used in this research includes several stages, including data analysis, software design, system implementation, testing, and application maintenance. The application of the MVC architectural pattern supports fast website development. This research produces a Trading Term Application, to help record and store trade agreement data, thereby increasing efficiency and order in business cooperation. The results of black box testing show that the Trading Term Application developed has passed a series of tests well and has achieved 100% success in the functional tests carried out.

Keywords: Application Trading Terms; Skeleton Pumpkin; Web Application; Python.

1. Introduction

Advances in global information technology have brought great benefits and made people's daily lives easier. Agencies or companies need to be able to access and manage data quickly, accurately, and efficiently. Therefore, having an information system that can organize information and data related to cooperation is a rational and valuable need. Information systems are widely known as applications that can help data analysis. The main goal of an information system is to convert initial data into helpful information for an agency or organization [1] like PT. XYZ, one of the largest retail companies in Indonesia with more than 20 years of operation and 14,300+ outlets, faces complexities in data management related to trade with suppliers. To ensure fulfillment of trade terms agreed with suppliers, PT. XYZ utilizes the Trading Term application. This application helps companies ensure that the costs associated with trading terms are optimized for the trading process and increase operational efficiency.

The importance of data processing and information presentation is also in focus, especially when organizations have large volumes of information. Therefore, organizations need an efficient information system to help manage and present information accurately and effectively. Information systems are tools that support the organization and analysis of data, aiming to produce helpful information for the company [2]. Overall, trading applications and information systems have a vital role in maintaining operational order and efficiency, especially in managing trade terms with suppliers or customers. Thus, it can be concluded that the development of information technology has positively impacted the ability of retail companies to manage data and information to support their collaboration with business partners.

There are still many companies that need help managing data well. They still record all activities, such as sales, purchases, inventory reports, and especially cooperation agreement reports, so management becomes complicated, resulting in difficulties compiling reports [3]. In addition, companies that have yet to be able to properly manage cooperation agreement documents experience difficulties finding complete data regarding the cooperation agreements they have executed [4]. Companies often carry out cooperation agreement transactions in the economic sector, so if they are not agreed upon clearly and in writing, they can cause disputes or problems.

System development was based on the Model, View, Controller (MVC) architecture and the Flask Framework, a web framework supported by the Python programming language, was used. Flask is categorized as a micro-framework because it does not require libraries or other tools. Flask uses Werkzeug and Jinja Template Engine as dependencies [5]. As a micro-framework, Flask allows the creation of program code with simple syntax. Using Flask, developers can easily select the components used in application development [6]. Most common functions and components, such as database validation and forms, are not automatically integrated. However, these functions and components have been prepared by third parties. By utilizing extensions, Flask can create the impression that these components and features are as if they had been applied directly by Flask itself [7].

Trading Term application development also utilizes PostgreSQL as a database. The PostgreSQL database is an open-source Relational Database Management System (RDBMS) technology that supports most SQL queries and provides complex query features, concurrent control, triggers, views and additional function extension features, data types, and procedural languages [8]. Making PostgreSQL a flexible and sophisticated solution for relational data management. The resulting Trading Terms application aims to make it easier for employees to record and store trade agreement data to increase efficiency and regularity in collaboration with business partners.

2. Research Method

The research method used in building the Trading Term application is the waterfall method. The Waterfall method is one of the earliest approaches in the Software Development Life Cycle (SDLC) to develop software [9]. The waterfall method is suitable for developing Trading Term applications because it makes it easier to work on applications where the process is carried out in stages and systematically so that the sequence of work processes is orderly. This research consists of stages involving planning, analysis, design, implementation, and testing. Figure 6 shows the stages of the waterfall method as follows.

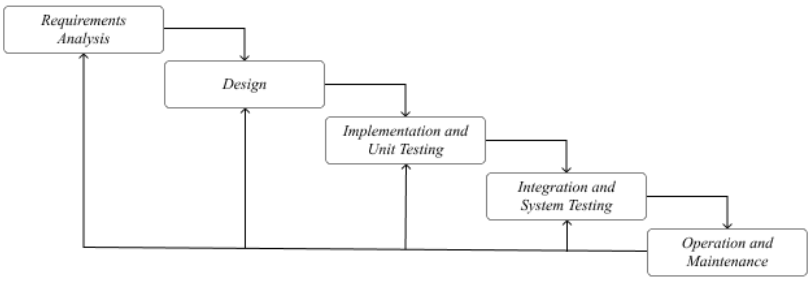

Figure 1. Waterfall Model

Requirements Analysis is the stage where data is analyzed to suit user needs. This includes collecting data for application development, which suppliers can obtain [10]. This needs analysis aims to explore the needs needed in the design phase through documents and other resources that support finding solutions to existing problems in the trading term. This process includes determining the main components, such as the software platform, programming language, database system, and technology to be implemented, all according to user needs. Trading Term application development will be website-based, using Python programming language, PostgreSQL database, and DBeaver tools with Flask technology. Software design involves creating a plan and structure for application development, including interface design, software architecture, and coding procedures. The aim is to provide a comprehensive overview of the tasks that must be carried out and assist in preparing hardware requirements. In developing the trading term application, the system design uses Unified Modeling Language (UML), with Use case Diagrams and Activity Diagrams to illustrate processes and interactions in the system. Implementation and Unit Testing is the stage where programs and databases are created based on previously designed designs. Each program module is tested using unit testing to ensure its functionality. The goal is to implement the design made into program code by dividing the software into small modules that will be combined in the subsequent integration stage. In developing the Trading Term application, the design that has been prepared is implemented in the Python programming language using the Flask Framework, and the database uses PostgreSQL with DBeaver tools. The implementation process is carried out using the Visual Studio Code IDE. Integration and System Testing involves testing the entire system to ensure it functions properly and meets specifications. This testing includes functional and non-functional testing, such as performance, security, and system reliability. In developing the Trading Term application, testing was carried out by Quality Assurance (QA), including black box testing, which focused on application functionality. The testing results will be compiled as a report, and if there are bugs or errors, QA will return the application to the developer for repair. Operation and Maintenance are the stages of application implementation, routine Maintenance, improvement, evaluation, and application development based on feedback to ensure the system can continue to run and develop according to its needs and functions [11]. The goal is to run the application system that has been created, ensure the system runs well, carry out regular Maintenance, and make improvements and developments according to needs.

3. Result and Discussion

3.1 Results

Development of the Trading Term application was built on a website using the Model View Controller system architecture; the MVC (Model, View, Controller) concept is an architecture where the system processes are separated into three parts. The first part is the Model, which contains database operations; the second is the View, which includes the interface; and the third is the Controller, which contains application logic and managing the View and Model [12]. This concept makes it easy for programmers to organize database queries, organize code, validate data, and maintain security. There are 3 types of components that build an MVC in an application.

1) Model, is the part that is responsible for managing data and is directly related to the database. In the Trading Term application, the model is named as DAO. The DAO section contains code that manages the database process.
2) View, which is part of the MVC architecture which is tasked with displaying information and presenting data in graphical form. In a Trading Term web application, this section uses an HTML template file, which is

managed by the controller. View functions to receive and present data to the user. This section does not have direct access to parts of the model.

3) Controller, is the part that regulates the interaction between the model part and the view part, the function of the controller is to receive requests and data from the user and then determine what will be processed by the application. In the Trading Term application, the controller is the part that holds all the process paths that connect the user with the display or display with the database.

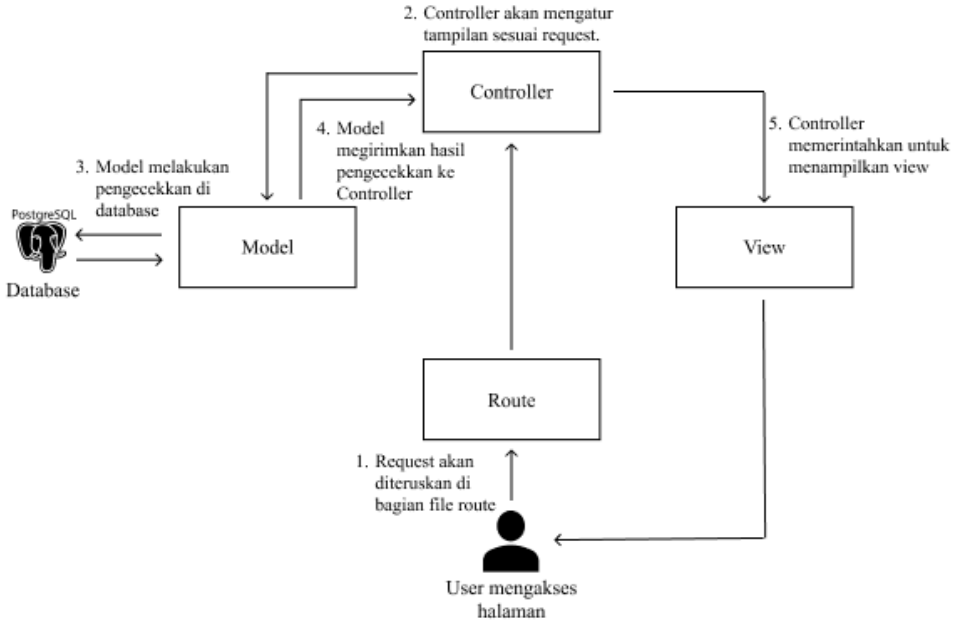

Figure 2. System Architecture

When the user opens the application, the controller will respond to a page to be displayed. Then, the user inputs the appropriate data in the form provided. After completing the data entry, the user saves the data. When the save button is pressed, the application will validate the data to be saved. The system requests the controller after the form data is filled in and validated correctly. The controller receives the request and sends it to the model section, which will search for and process the requested data in the database. This research obtained results through the Trading Term application, a web-based application that can be accessed only via the internet or internal company network. This application development uses Python technology with support for the Flask framework as a web service, PostgreSQL as a database, and HTML for the interface. FlaskFlask has a simple and easy-to-understand syntax, allowing developers to build applications quickly. In addition, Flask is light and flexible, making it suitable for application development.

Program Code 1. Controller.py

```
  $http.post('/drafttradingtermprinsipal/detail', data)
  .then(response => response.data)
  .then(res => {
  detail = res[0]['nilai']
  resolve('selesai1');
});
```

Program code 1 is part of a web application using Flask. In line 1, this route is defined as responding to requests with the GET and POST methods on the URL /drafttradingtermprincipal/. The `@login_required` program code is used to set up a web route protected by authentication so that logged-in users can only access this route. The last line of program code functions to render the "Draft Trading Terms Principal" page using the HTML template that has been prepared. By using this code, the application ensures security by requiring users to log in before accessing it so that only authorized users can use the available features according to the roles and access rights that have been assigned.

Program Code 2. Trading Terms

```
$http.post('/drafttradingtermprinsipal/detail', data)
  .then(response => response.data)
  .then(res => {
```

```
  detail = res[0]['nilai']
  resolve('selesai1');
});
```

This code functions in an HTML file, which sends an HTTP POST request to a specified endpoint, namely /drafttradingtermprincipal/detail, with data specified by the "data" variable. Once the request is sent, the code handles the server's response. Then, the value from the response will be retrieved using JavaScript. Finally, the value will be stored in the "detail" variable on the page. By using asynchronous code, the application does not stop the execution of other code running simultaneously. This allows applications to remain responsive and efficient in handling network requests or operations that take a long time, such as HTTP requests. Additionally, using promises makes it easier to handle responses from such requests in a structured and easy-to-understand way.

Program Code 3. LOV

```
  $scope.lovTop = ()=> {
  $http.get('/drafttradingtermprinsipal/lovTop')
  .then(response => response.data)
  .then(res => {
  $scope.dataTop = res
  angular.element("#modalLovTop").modal('show');
  });
  }
  $scope.selectDataTop = (v)=> {
  $scope.datas.ttNew.ttTopNew = v.kd_b
  $scope.datas.ttNew.ttTopHariNew = v.lama
  $scope.datas.ttNew.ttTopDescpNew = v.cara_b
  angular.element("#modalLovTop").modal('hide');
}
```

In program code 3 using AngularJS, the $scope.lovTop function retrieves data from the server using an HTTP GET request to /drafttradingtermprincipal/lovTop. This code allows the application to recover data from the server asynchronously, so it does not stop the execution of other code. The received data is then stored in the $scope.data variable and a modal with the ID modalLovTop on line 6 is displayed to display the data to the user. The $scope.selectDataTop function in line 9 handles selecting data from the list of values (LOV) displayed in the modal. When the user selects data, the selected values are saved into the $scope. data.ttNew variable and the modal is hidden. Using AngularJS makes data management and user interface interaction easier.

Program Code 4. Loop Table LOV

```
  <tr ng-repeat="v in dataTop | filter: cariTop" ng-click ="selectDataTop(v)">
  <td>{{ '{{ v.kd_b }}' }}</td>
  <td>{{ '{{ v.cara_b }}' }}</td>
  <td>{{ '{{ v.lama }}' }}</td>
  <td>{{ '{{ v.flag_tepat }}' }}</td>
</tr>
```

Program code 4 is part of an HTML template that uses AngularJS. Using the "ng-repeat" directive in line 1 allows iterating through each item in the 'data' array so that the table view will automatically update according to the array's contents without the need to update the code manually. The 'search' filter filters data based on the user's search criteria so that users can easily filter the data displayed according to the search criteria entered. Each item in the array is represented by the variable 'v.' The "ng-click" directive is used to specify the action to take when the user clicks on the form, in this case, calling the selectDataTop(v) function with the item 'v' as an argument. This increases the interactivity and flexibility of the user interface. By using AngularJS, this code is reusable and can be reused in different parts of the application with similar functionality.

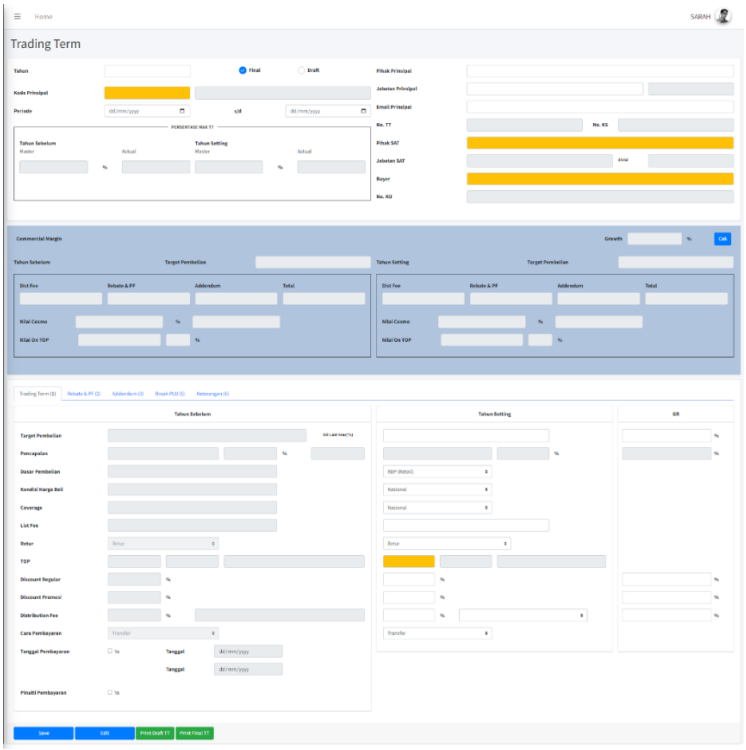

Figure 3. Application display

Figure 3 displays the Trading Terms page; users can fill in the data in the form. The data is the desired year. After that, the date period is filled automatically according to the year data entered by the user. Then, the user can select the principal code by clicking the yellow List of Values (LOV). After the user selects the principal code, the other forms will be filled in automatically if the data is in the database. Users can edit the data in the form according to their needs. Then, the user can save the data changes by clicking the save button below. This application can efficiently manage data and cooperation agreement documents with suppliers. Users can easily access, store, and manage trading information quickly and efficiently. With this application, companies can optimize the process of managing trade data, cooperating with suppliers, and maintaining order in preparing reports and documentation of cooperation agreements. This reduces the potential for errors and conflicts due to a lack of regularity in data management. Using the Flask framework allows fast and efficient application development thanks to the ease of configuration and the variety of libraries available. Using the waterfall development method, the application development process can be well structured through the stages of analysis, design, implementation, testing, and maintenance. This can improve the quality of the application. Testing was conducted on the Trading Term Application by the company's Quality Assurance (QA) to test functionality. This test uses BlackBox Testing because it tests the smoothness of the software and programs created without testing the design and knowing the program code used [13]. The main advantage of using BlackBox Testing is that testing can be carried out without the need for in-depth knowledge of the programming language used and allows testing to be carried out from the user's perspective to identify inconsistencies and ambiguities. This also creates a positive dependency between application developers and testers in the testing process [14].

Table 1. Trading Term Application Testing

No	Description	Expected results	Test result	Status
1.	The year period must be filled in at least 1 year in the past and a maximum of 3 years in the future	The system will reject and then display an alert if the form filling is not appropriate	Already appropriate	Valid
2.	Displays the LOV display on the principal code if the year period input is appropriate	The system refuses and then displays an alert and will not display the LOV display if the year period is still empty and does not match	Already appropriate	Valid

3.	Leaving the input form blank and then clicking the "Save" button.	The system rejects and displays an alert.	Already appropriate	Valid
4.	Other forms will be filled with data automatically after selecting data from the principal code LOV.	The form will be filled with data automatically if the data is in the database, and the form will remain empty if there is no such data in the database.	Already appropriate	Valid
5.	Fill out the year form appropriately	Forms for periods 1 and 2 will be filled in automatically if the years entered are correct.	Already appropriate	Valid

The black box testing results show that all stages of application testing have been successfully implemented thoroughly and by the user's functional needs without the need for internal knowledge about the structure or program code. This confirms that the app has successfully passed testing and that all features are working correctly from the user's perspective. Therefore, the success of the whole experiment reached a rate of 100%, confirming that the software operates well and meets the initial requirements.

3.2 Discussion

In developing a web-based Trading Term application using the Flask framework at PT. XYZ, several important aspects need to be discussed. This discussion includes an evaluation of the success of the project, challenges faced during development, and implications and recommendations for the development of similar applications in the future. This research succeeded in developing a Trading Term application that meets the needs of PT. XYZ in managing trade agreement data with its suppliers. The application has a simple and intuitive user interface, making it easy for users to record, store, and update trade agreement data. Test results show that this application functions well and meets the specified functional requirements. The success of this project is reflected in the high level of user satisfaction and increased operational efficiency observed by PT. XYZ after using this application. Although the project was successful, several challenges were faced during the application development. One of the main challenges is time and resource management. Application development requires significant time and effort, especially regarding requirements analysis, software design, and system implementation. Additionally, integration between various application components is a challenge, mainly due to the complexity of business processes complexity of business processes. However, with the support and cooperation of the development team, this challenge was overcome. This research implies that developing web-based applications using the Flask framework can significantly benefit companies in managing data and information. This application can help improve operational efficiency and regularity in business collaboration with trading partners. A recommendation for future development of similar applications is to pay more in-depth attention to user needs and involve them in every stage of development. Additionally, it is crucial to continuously update and improve the app according to user feedback and the latest technological developments. Development of a web-based Trading Term application using the Flask framework at PT. XYZ is an essential step in increasing the company's operational efficiency. This application succeeded in meeting the needs of PT. XYZ manages trade agreement data with its suppliers, significantly benefiting the company. Although several challenges were faced during development, the success of this project shows the great potential of using information technology to support business activities. Thus, it can be concluded that the development of this application has succeeded in achieving the stated objectives and made a positive contribution to PT. XYZ.

4. Related Work

Research conducted by Salma Nada Safira *et al*. (2021) created a valuable application to facilitate the storage and management of data related to cooperation agreements between two parties. The method used by the author is the prototype method in designing a collaboration system. This method comprises requirements gathering, prototype development, prototype evaluation, system coding, system testing, system evaluation, and documentation [15]. The results of this research are a collaboration system application that allows users to enter and view data, including MoU submissions, PKS submissions, MoU cooperation, cooperation agreements (PKS), cooperation implementation, and monitoring and evaluating cooperation. This application can also print data in recapitulation reports. Previous research has a similar case relationship,

namely discussing the application of cooperation agreements, which provide quick and accessible information. In this research, the system can store documents well so that they become a reference in research to implement a system that can facilitate the archiving process well.

Research conducted by Yuny Tamariska Bota and Nina Setiyawati (2022) In their study, the authors aim to build a web-based business intermediary information system that can provide transaction information such as purchase reports and supplier invoices. This information system was developed using the waterfall model method and the Flask framework, which uses Python as the programming language. This research results in a website-based cooperation information system with the main features for managing Cooperation Agreement (PKS) data, MoUs, monitoring cooperation, implementing cooperation, and viewing and printing final summary reports. Previous research has the same technological relationship using the Flask Framework to build web-based applications by utilizing several Flask functions, such as managing data [16]. In this research, the Flask Framework, a Python micro-framework, is used. Flask does not include standard components like form validation, databases, or others. However, Flask supports extensions to add this functionality using the Python programming language [17]. This more straightforward and easier-to-read programming language can make it easier for authors to implement it on systems. Hence, it becomes a reference for research on implementing the Flask Framework.

The research Implementation of the Flask Framework in Developing Reporting Modules for Helpdesk Information System Applications. In this research, the author focuses on increasing the efficiency and effectiveness of business processes at PT. The author uses a waterfall model, starting with problem identification and ending with system testing. The result is a website repository for information related to company issues [18]. the same technological relationship, namely, using the Flask Framework as the main framework for building websites that focus on the backend. In this research, the use of the Flask Framework is focused on the backend part, suitable for data management, so it becomes a reference for research on the application of the Flask Framework.

The three studies presented have the same focus on using information technology, especially Flask, to build systems that can store, manage, and present information related to collaboration between institutions or companies and their business partners. These studies aim to improve operational efficiency and regularity in storing data on cooperation agreements, purchases, and final summary reports. With a focus on backend management and critical features such as document storage and transaction reporting, the results of this research provide an essential contribution to the development of information systems that simplify collaboration processes and data management in companies. In contrast to previous research, which used the PHP programming language in the creation process, the author chose Python. This is because Python has a versatile library for various applications, from statistics to deep learning. Python's advantage lies in its flexible and easy-to-use properties in various application development.

5. Conclusion

Based on the discussion above, it can be concluded that using the Flask framework in developing the Trading Term application can provide convenience for developers. This is because the Flask framework provides many ready-to-use libraries. With this library, developers can work more quickly and efficiently in building applications because they don't need to start from scratch and can take advantage of the existing functions in the framework. Thus, using the Flask framework can speed up the Trading Term application development process and increase the productivity of application developers. With the Trading Terms application, companies can more effectively manage and accurately present information and data. Trading applications and information systems are vital in maintaining operational order and efficiency. This is especially true for retail companies managing trade terms with suppliers, where information systems can help ensure more efficient fulfillment of agreed terms. Thus, this application becomes a precious tool for companies in increasing productivity and optimizing business processes.

References

[1] Muryono, L., *et al*. (2023). Perancangan Sistem Informasi Kerjasama Pada Fakultas Teknik Universitas Sebelas Maret Surakarta. Retrieved from https://journal.polhas.ac.id/index.php/imaging

[2] Olva, M., Dwiputri Permatasari, R., Majid, S., Syair, P., & Suganda, A. (2021). Pemanfaatan Dasbor pada Pemantauan Data Transaksi Penjualan. *Journal of Engineering, Technology, and Applied Science, 3*(1), 1–15. https://doi.org/10.36079/lamintang.jetas-0301.188

[3] Rahadi, D. M., Kusnadi, K., Suwandi, S., & Pranata, S. (2023). Sistem Aplikasi Pencatatan Transaksi Penyewaan Mesin Fotokopi Pada Pt. Radiance Cirebon Berbasis Web. *Jurnal Manajemen Sistem Informasi*, *1*(2), 46-54. https://doi.org/10.51920/jurminsi.v1i2.147.

[4] Satria, B., & Romadhoni, R. (2021). Aplikasi Dokumentasi Elektronik Memorandum Of Understanding (MoU) Dan Nota Kesepahaman Bersama Di Jurusan Kemaritiman Polbeng. *INTECOMS: Journal of Information Technology and Computer Science*, *4*(2), 240-246. https://doi.org/10.31539/intecoms.v4i2.3016.

[5] Al, M., Deva, F., Putra, S., & Susetyo, Y. A. (2023). Jurnal Teknologi Sistem Informasi dan Aplikasi Pembangunan Stock Monitoring Application untuk Pembuatan Laporan Goods in Transit Menggunakan Flask, *6*(2), 54–62. https://doi.org/10.32493/jtsi.v6i2.29195

[6] Todingbua, D., & Setiyawati, N. (2022). Pengintegrasian Penyimpanan Google Drive Pada Pembangunan Aplikasi Monitoring Project Menggunakan pyDrive. *JURIKOM (Jurnal Riset Komputer), 9*(4), 800. https://doi.org/10.30865/jurikom.v9i4.4452

[7] Ghimire, D. (2020). Comparative study on Python web frameworks: Flask and Django. https://urn.fi/URN:NBN:fi:amk-2020052513398.

[8] Indrayana, I. M., Sukarsa, I. M., & Wibawa, K. S. Text Similarity Measurement using PGTRGM Module Function PostgreSQL. *Jurnal Ilmiah Teknologi dan Komputer*, *3*(1), 802-809.

[9] Wahid, A. A. (2020). Analisis metode waterfall untuk pengembangan sistem informasi. *J. Ilmu-ilmu Inform. dan Manaj. STMIK, no. November*, 1-5.

[10] Wibowo Putri, A. A., & Susetyo, Y. A. (2022). IMPLEMENTATION OF FLASK FOR STOCK CHECKING IN DISTRIBUTION CENTER & STORE ON MONITORING STOCK APPLICATION IN PT. XYZ. *Jurnal Teknik Informatika (Jutif), 3*(5), 1265–1274. https://doi.org/10.20884/1.jutif.2022.3.5.334

[11] Kase, J. S. P., & Saian, P. O. N. (2023). PENGEMBANGAN APLIKASI REPORTING PENENTUAN KUANTITAS MAKSIMUM PADA SISTEM APLIKASI OPERASI PERUSAHAAN XYZ MENGGUNAKAN FLASK FRAMEWORK. *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika), 8*(4), 1288–1299. https://doi.org/10.29100/jipi.v8i4.4231.

[12] Alkaff, M., Radam, I. F., & Chandra, W. (2021). Penerapan Pattern MVC (Model View Controller) dalam Pengembangan Aplikasi Identifikasi Jam Puncak Arus Lalu Lintas pada Simpang Lima. *JUSTINDO (Jurnal Sistem dan Teknologi Informasi Indonesia)*, *6*(2), 91-97. https://doi.org/10.32528/justindo.v6i2.4381.

[13] Wijaya, Y. D., & Astuti, M. W. (2021). Pengujian Blackbox Sistem Informasi Penilaian Kinerja Karyawan Pt Inka (Persero) Berbasis Equivalence Partitions. *Jurnal Digital Teknologi Informasi*, *4*(1), 22-26. https://doi.org/10.32502/digital.v4i1.3163.

[14] Ningtyas, D. F., & Setiyawati, N. (2021). Implementasi Flask Framework pada Pembangunan Aplikasi Purchasing Approval Request. *Jurnal Janitra Informatika dan Sistem Informasi, 1*(1), 19–34. https://doi.org/10.25008/janitra.v1i1.120.

[15] Safira, S. N., & Rahayu, T. (2021). Perancangan Sistem Informasi Kerjasama pada Fakultas Ilmu Komputer UPN Veteran Jakarta dengan Framework Laravel. In *Prosiding Seminar Nasional Sains dan Teknologi Terapan* (Vol. 4, pp. 335-344).

[16] Bota, Y. T., & Setiyawati, N. (2022). Pengembangan Sistem Informasi Perantara Bisnis Menggunakan Framework Flask. *Journal of Information Technology Ampera*, *3*(2), 79-93.

[17] Santoso, B. B., & Saian, P. O. N. (2023). Implementasi Flask Framework pada Development Modul Reporting Aplikasi Sistem Informasi Helpdesk di PT. XYZ. *Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi)*, *7*(2), 217-226. https://doi.org/10.35870/jtik.v7i2.718.

[18] Junaidi, S., Devegi, M., & Kurniawan, H. (2023). Pelatihan Pengolahan dan Visualisasi Data Penduduk menggunakan Python. *ADMA : Jurnal Pengabdian dan Pemberdayaan Masyarakat, 4*(1), 151–162. https://doi.org/10.30812/adma.v4i1.2963.

