similarity_133.docx

by . Turnitin

Submission date: 26-Jun-2026 01:09PM (UTC+0900)
Submission ID: 2989624344
File name: similarity_133.docx (969.9K)
Word count: 2858
Character count: 16722

Development Of A Press Machine Loan Information System Using The FIFO (First In First Out) Algorithm (A Case Study Of A Garment Company In Bandung)

Abstract

One gaming company continues to manage the borrowing of press machines traditionally through paper forms and spreadsheet documentation. This approach frequently results in problems, including scheduling conflicts, recording inaccuracies, and ambiguous borrowing sequences. This study seeks to resolve these issues by creating a web-based information system for borrowing press machines that utilizes the FIFO (First In First Out) algorithm. The research employs a descriptive-analytic qualitative approach, while the system development adheres to the Agile (Scrum) framework for enhanced flexibility and adaptability.
The system provides functionalities such as an automated FIFO queue, online borrowing, real-time machine status monitoring, notifications, and usage analytics. Testing results indicate that implementing FIFO significantly reduces scheduling conflicts, improves queue fairness, and expedites the borrowing process. This research not only provides practical benefits for the organization but also advances the scientific application of queuing algorithms in industrial asset management information systems.

Keywords : *Information System, Press Machine, FIFO, Agile Scrum, Asset Management*

INTRODUCTION

A garment manufacturer in Bandung considers the management of the press machine a vital element that demands high efficiency and precision in manufacturing operations. An internal audit conducted in 2023 revealed that the traditional paper-based and spreadsheet approach for renting press machines had a 15% error rate in recording, resulting in scheduling conflicts.

The primary challenges encountered include the lack of a definitive queue management system, inaccuracies in human record-keeping, and the inability to monitor machine condition in real time. The antiquated system operates on a "first-come, first-served" basis, with no organized documentation, resulting in confusion and scheduling issues with machine utilization. A digitized and automated information system is required to efficiently record, organize, and manage machine borrowing to resolve these issues. The FIFO (First In First Out) algorithm has been chosen as a solution due to its capacity to provide equity and clarity in queue management. FIFO ensures that the initial request is addressed first, rendering it appropriate for settings with a constant influx of borrowing requests, such as manufacturing facilities. Prior research by (Yulianto, 2021) demonstrated that applying FIFO in public service wait systems could enhance efficiency by up to 60%.

A. System Development

The system development process entails creating a new system to entirely replace the old one or enhancing an existing one. The system analyst is integral to the development and design of systems, focusing on creating a highly functional system that meets end-user requirements (Saputra et al., 2025).

B. Fundamental Concept of a System

A system comprises a set of interrelated components that collaborate to accomplish defined objectives. It comprises elements that engage with and rely on each other (Nasrullah et al., 2024).

C. Fundamental Concept of Information

Information denotes data that has been processed to impart significance for users in decision-making (Rizky Fawzy et al., 2024). It is the outcome of data processing, augmented with

context, significance, and particular aims, rendering it valuable to the recipient.

D. Asset Management Information System

An Asset Management Information System (AMIS) is a technological application utilized to oversee the life cycle of an organization's assets, encompassing borrowing, purchase, utilization, maintenance, and disposal (Hasanah & Purnomo, 2022).

E. FIFO (First In, First Out) Algorithm

The FIFO (First In First Out) algorithm is a data management technique in which the data entered first is processed or eliminated first. This notion is extensively applied in queue systems, buffers, process scheduling, and asset borrowing systems (Aryani & Ali, 2025).

F. Laragon

Laragon is a packaging program that provides a streamlined, efficient local development environment for web apps. It consolidates multiple fundamental components, including Apache, MySQL, PHP, and Node.js, into a singular, user-friendly installation package. Laragon's benefit is its ability to set up a local server quickly, quickly set up virtual hosts, and offer robust integration with diverse development tools (Reghan Supremadinata et al., 2026).

G. Draw.io

Draw.io is a platform designed for creating UML diagrams online. It boasts a highly responsive UI and interacts with Google Drive for file storage, rendering Draw.io an efficient choice for UML diagram production with a reduced turnaround time (Agus Waryanto & Rohmad Haryadi, 2022).

H. Unified Modeling Language (UML)

The Unified Modeling Language (UML) is a standardized visual modeling language used for the design and documentation of software systems, especially object-oriented systems. (Kamila, 2025) The document encompasses various diagram types: the Use Case Diagram, which delineates primary elements such as actors and use cases, facilitating the analysis and design of system requirements; the Activity Diagram, which portrays the progression of processes or activities from initial actions to decision-making and final steps; the Sequence Diagram, which conveys the messages exchanged among use cases in a temporal sequence; and the Class Diagram, a static representation that illustrates the system's structure through classes, attributes, methods, and inter-class relationships.

I. Black-box testing

Black-box testing is a methodology that assesses software functioning without any insight into its internal architecture. Testing is performed from the user's viewpoint, concentrating exclusively on the system's input and output. (Waryanto & Haryadi, 2022)

J. Laravel

Laravel exhibits a more refined and articulate design. It provides several functions that assist PHP developers in efficiently and swiftly creating web applications, including robust tools and architecture. The Model View Controller (MVC) is a software development architectural pattern that divides the application into three components: Model, View, and Controller. This methodology organizes apps to enhance code reuse and facilitate maintenance. In the MVC pattern, the application's components are categorized into three segments: the Model encapsulates the data structure and business logic; the View functions as the output representation of the Model or the user interface perceived by users; and the Controller is tasked

with receiving user input and converting it into commands for the Model and/or view.

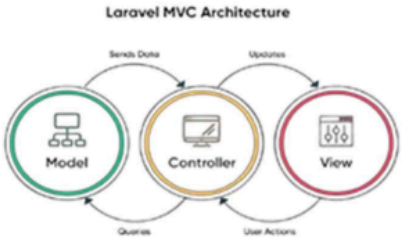

Figure 1. MVC Workflow

This study seeks to develop a web-based information system for the borrowing of press machines using the FIFO algorithm. It will offer functionalities including automatic queue registration, real-time status monitoring, and historical report generation. The system is meant to be flexible and adaptable to evolving user requirements through an Agile (Scrum) development methodology.

METHOD

Research methods

This study employs a descriptive-analytical framework with qualitative methodologies to systematically gather, analyze, and report data on the press machine borrowing system at PT. Trisco TAM. The selected methodology seeks to delineate the existing condition of the traditional system, pinpoint problems, and assess user requirements before system construction. The data types and sources include interviews with maintenance administrators, production personnel, and maintenance technicians, as well as firsthand observations of the press machine borrowing procedure, which record the durations and faults observed. A literature review is performed to collect data and information from books, online articles, journals, and other pertinent sources, which will serve as reference material for this study.

Development Method

The Agile (Scrum) development methodology is a component of agile methodologies intended to facilitate iterative processes and phases in the ever-evolving domain of business system development. This methodology facilitates a more organized system development by engaging users from the earliest stages through execution. The process commences with the planning phase, during which initial user requirements are gathered by identifying issues, formulating user stories, establishing software usage objectives, determining requisite information, and analyzing application necessities. Subsequently, during the design phase, the system is depicted using UML models, including use case and activity diagrams, thereby enhancing developers' comprehension.

In the coding phase, the development team immediately implements the design, which entails building a MySQL database and crafting the user interface. The system is constructed on PHP with the Laravel 11 framework, augmented by HTML, CSS, JavaScript, and Bootstrap for the frontend. Laragon functions as the local web server, and Visual Studio Code operates as the

code editor. The FIFO algorithm is implemented in the borrowing module to manage the queue by request timestamps efficiently, ensuring First In First Out. The testing step signifies the apex of this research, enabling the results to be subjected to trials for a comprehensive assessment of the web application's design, functionality, and performance.

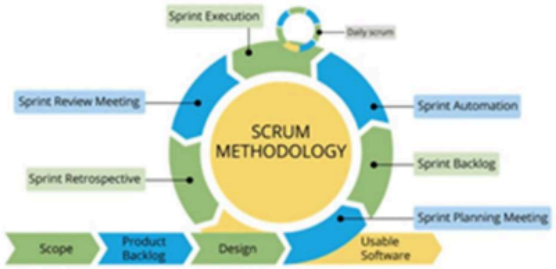

Figure 2. Agile Method (Scrum)

RESULT AND DISCUSSION

Observations and conversations with the Maintenance Team at a garment manufacturer in Bandung reveal that the existing press machine borrowing system functions customarily and encounters many significant challenges:

1) Manual Recording Process: Operators complete paper forms that are documented in a notebook and Excel, requiring 10-15 minutes per transaction with an error rate of 12-16%.
2) Absence of Queue System: The "first come, first served" premise, devoid of a definitive queuing mechanism, leads to 4-5 schedule conflicts daily.
3) Lack of Real-Time Monitoring: Machine status can solely be assessed via Excel or direct communication, resulting in frequent delays in status updates.
4) Unstructured Documentation: Data is dispersed throughout five distinct Excel files without systematic backups, resulting in a significant risk of data loss.

The borrowing frequency is 24-25 transactions per day, with peaks between 08:00-10:00 (28%) and 13:00-15:00 (35%). A total of 15 press machines are distributed over 7 production lines.

a. Design

A Use Case is a diagram in Unified Modeling Language (UML) that depicts the interactions between actors (users or external systems) and the system under development. The Use Case Diagram delineates the anticipated features of a system through a graphical representation of the interactions between actors and the corresponding system. The following is the Use Case Diagram developed for the Press Machine Borrowing System for a garment industry in Bandung.

Figure 3. Use Case Diagram for Borrowing a Press Machine

b. Activity Diagram

The activity diagram also facilitates analysis of the use case diagram by delineating the players, required actions, and their timing. The following is the activity diagram contained within this system:

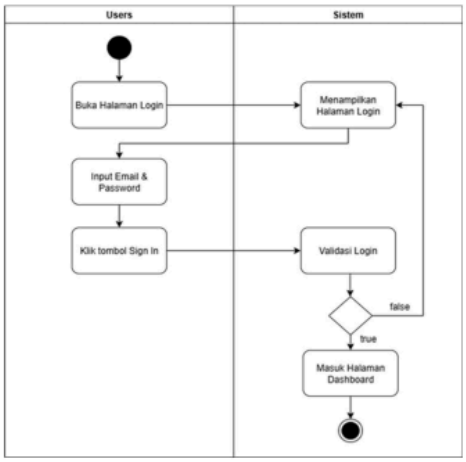

Figure 4. User Login Activity Diagram

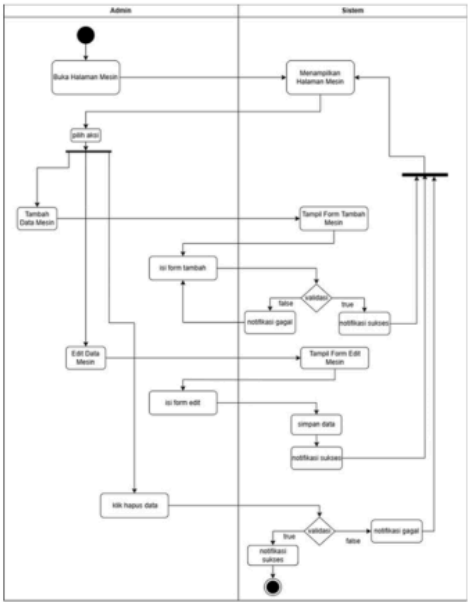

Figure 5. Machine Activity Diagram

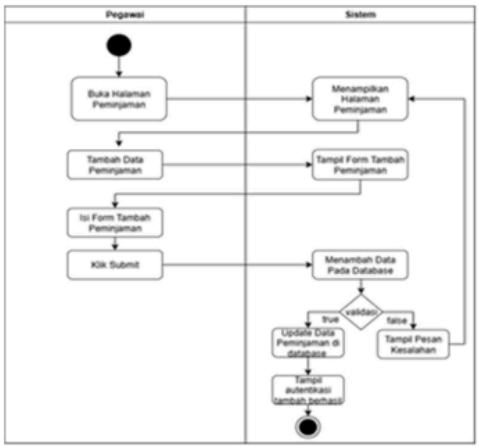

Figure 6. Borrowing Activity

Figure 7. Machine Return Activity Diagram

c. Sequence Diagram

The sequence diagram depicts the chronological flow of interactions among objects inside a system. The vertical lines in the diagram indicate the passage of time from top to bottom, and the horizontal arrows illustrate the communication between actors or objects. Consequently, the sequence diagram facilitates a comprehensive comprehension of the execution of a particular feature or use case from inception to conclusion.

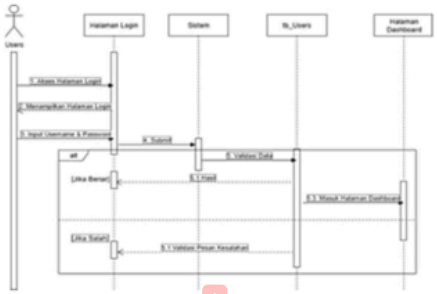

Figure 8. Login Sequence Diagram

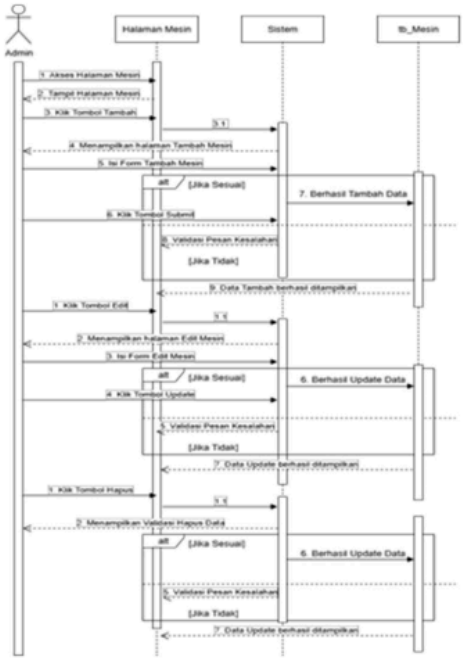

Figure 9. Machine Sequence Diagram

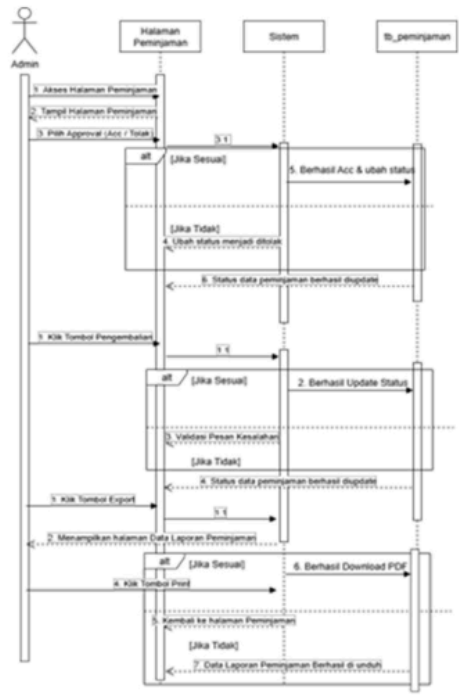

Figure 10. Sequence Diagram of Borrowing

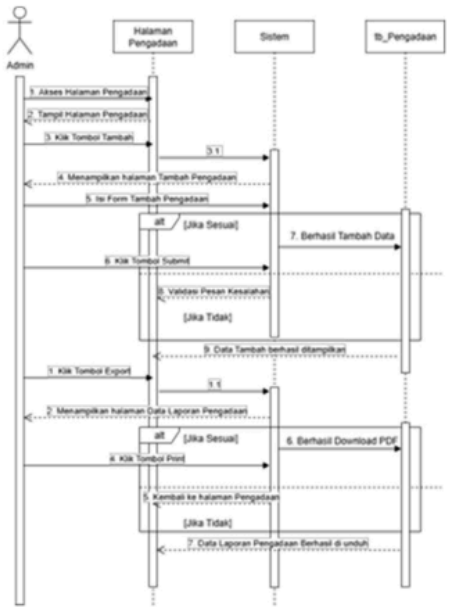

Figure 11. Machine Return Sequence Diagram

d. Class Diagram

The class diagram illustrates the static structure of a system. The graphic illustrates the classes within the system, including their attributes, methods, and interrelationships. A class diagram enables the visualization of the interconnections among the system's components. In a class diagram, the principal class is the central class within the system.

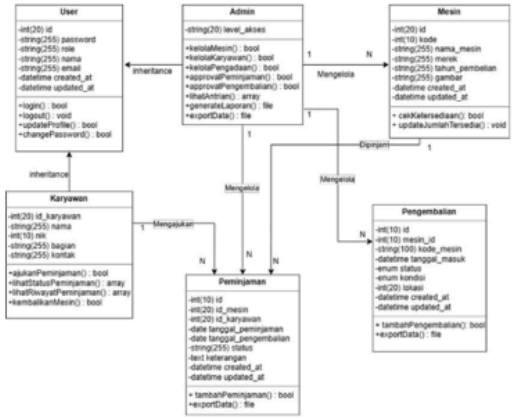

Figure 12. Class Diagram

e. Database Implementation

The discourse on the table structure in the database outlines the comprehensive design of all tables within the program's database. This program comprises a single database that encompasses multiple tables, including:

Figure 13. Users Table

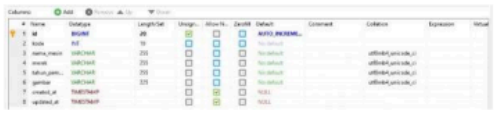

Figure 14. Machine Table

Figure 15. Employee Table

Figure 16. Loan Table

Figure 17. Return Table

f. Interface Implementation

The implementation phase in information system development is the phase during which the designed system is converted into a functional, usable application. The final results are shown in the system below:

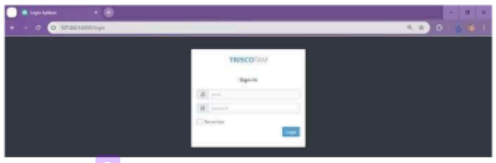

Figure 18. Users Login Page

Figure 19. Dashboard Menu Page

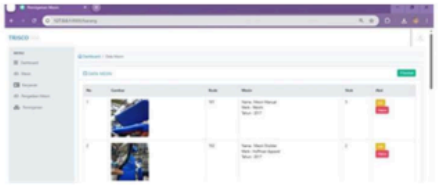

Figure 20. Machine Menu Page

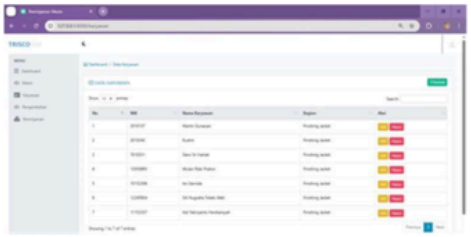

Figure 21. Employee Menu Page

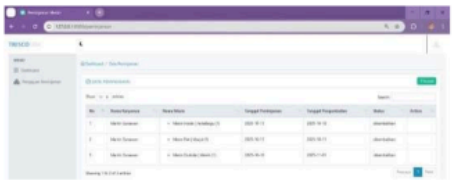

Figure 22. Machine Loan Page

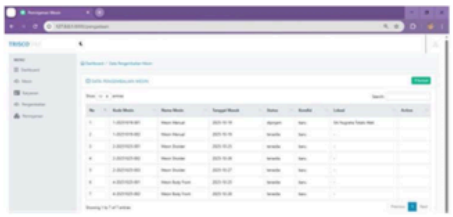

Figure 23. Machine Return Page

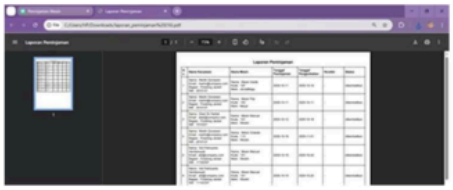

Figure 24. Data Export Results Page

g. System Testing

This application testing emphasizes Black-Box Testing to assess the system's functionality and analyze program execution outcomes. It guarantees that designated inputs will activate the appropriate processes and yield outputs that conform to the design.

Table 1. Login Functionality Test Results

No	Testing Scenario	Test Case	Expected results	Test Results	Status
1	Login with valid credentials (Admin)	Username: admin Password: admin123	Successfully logged in and directed to the admin dashboard	The system successfully authenticates and displays the admin dashboard.	✓ Valid
2	*Login with valid credentials (Employee)*	*Username*: karyawan01 Password: karyawan123	Successfully logged in and directed to the employee dashboard	The system successfully authenticates and displays the employee dashboard.	✓ Valid
3	Login with the wrong username	Username: adminSalah Password: admin123	Login failed, error message appears	The system displays the message "Incorrect username or password."	✓ Valid
4	Login with incorrect password	Username: admin Password: salah123	Login failed, error message appears	The system displays the message "Incorrect username or password."	✓ Valid
5	Login with blank fields	Username: (kosong) Password: (kosong)	Form validation, unable to submit	The system displays the message "Field is not empty."	✓ Valid

Table 2. Machine Data Management Test Results

No	Testing Scenario	Test Case	Expected results	Test Results	Status
1	Displays a list of machines	Admin accesses the Machine menu	The system displays a list of all machines with complete details.	The machine list is displayed with code, name, specifications, and status information.	✓ Valid
2	Adding new machine data	Admin fills out the form: - Code: MCP-004 - Name: Manual Machine - Specifications: 50 Tons - Status: Available	New machine data is successfully saved and appears in the list.	The new machine was successfully added to the database and appears in the machine list.	✓ Valid
3	Adding a machine with duplicate code	Admin fills out the form with the existing machine code	The system rejects and displays an error message.	The system displays the message "The machine code has been registered."	✓ Valid
4	Editing machine data	Admin changed the machine name MCP-004 to"Mesin Cuff"	Machine data successfully updated	Data changes are saved and displayed in the machine list.	✓ Valid
5	Deleting machine data	Admin deletes machine with code MCP-004	Machine data has been deleted from the system.	The machine was successfully removed from the database and does not appear in the list.	✓ Valid
6	Deleting the machine that is currently on loan	The admin tried to delete a machine with the status "Borrowed"	The system rejects the deletion.	The system displays the message "Unable to delete the machine currently on loan."	✓ Valid

Table 3. Results of Admin Machine Loan Testing

No	Testing Scenario	*Test Case*	Expected results	Test Results	Status

1	Displaying the loan list	Admin accesses the Loan menu	The system displays all loan applications with various statuses.	The loan list is displayed complete with information about the borrower, machine, and status.	✓ Valid
2	Approving the loan	The admin approved the loan application with the status "Pending."	The status changes to "Approved" and the machine is now "Borrowed."	Approval successful, status updated, notification sent to employee	✓ Valid
3	Refusing a loan	The admin rejected the application for a specific reason.	The status changes to "Rejected" and the reason is listed.	Rejection successful with reason saved	✓ Valid
4	Approval for returning the machine	Admin confirms return of machine	The machine status returns to "Available" and the next queue is processed.	Returns recorded, machine available, queue moving automatically	✓ Valid
5	Filter loans by status	The admin filters loans with the "Approved" status.	The system displays only loans with the selected status.	The filter works well, the data meets the criteria	✓ Valid

Table 4. FIFO Algorithm Test Results

No	Testing Scenario	Test Case	Expected results	Test Results	Status
1	Queue order based on application time	Three employees apply to borrow the same machine in sequence: Employee A: 8:00 AM - Employee B: 8:15 AM - Employee C: 8:30 AM	The queue is structured: A → B → C	The system arranges the queue according to the correct chronological order of submissions.	✓ Valid
2	Queue processing after return	The machine is returned, the first queue is processed automatically.	Employee A (first in line) receives a notification and his status changes.	The first queue is automatically processed, the notification is sent, and the status is updated.	✓ Valid
3	Real-time queue position	Employees see their position in the queue	The system displays the queue position and estimated waiting time.	Queue information is accurate and updated in real time.	✓ Valid
4	Queue Shift after Approval	Admin approves the first loan in the queue	The queue shifts, the next position becomes priority.	Automatic queue shifts, positions are updated for all users	✓ Valid
5	Validation has no special priority	Employees at different levels apply for loans	Everyone is treated equally, only based on the time of submission.	There is no priority based on position, only FIFO	✓ Valid

CONCLUSION

The research findings indicate that implementing a press machine information system using the FIFO algorithm at a garment firm in Bandung efficiently addresses the lack of a systematic queuing mechanism. The system autonomously regulates borrowing queues according to submission timestamps, thereby reducing

scheduling conflicts and recording inaccuracies. The notion of time-based prioritization is implemented equitably and transparently. The system offers real-time monitoring capabilities that enable users to monitor machine availability without frequent interactions with the administrator, thereby improving operational efficiency and alleviating administrative burden. The structured reporting feature also produces automatic borrowing reports in PDF format, complete with extensive historical data. This data facilitates the examination of machine utilization trends and informs strategic decisions for production asset management.

REFERENCES

similarity_133.docx

ORIGINALITY REPORT

6%	4%	1%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Kingston University	Student Paper
2	ojs.unpkediri.ac.id	Internet Source
3	Submitted to Texas A & M University, Kingville	Student Paper
4	www.softguide.com	Internet Source
5	Submitted to University of Central Lancashire	Student Paper
6	www.erawa.com.au	Internet Source
7	dspace.enset-skikda.dz	Internet Source
8	www.tutorialspoint.com	Internet Source

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		