



**SMART MANUFACTURING EXECUTION SYSTEM: REVOLUTIONIZING
INDUSTRY WITH IOT AND AI
(MODULE: FULL STACK WEBSITE)**

**UNDERGRADUATE THESIS
Submitted as one of the requirements to obtain
Sarjana Komputer**

By:

MOCHAMAD HERTANTO WICAKSONO

001202200070

**FACULTY OF COMPUTER SCIENCE
INFORMATICS STUDY PROGRAM
CIKARANG
AUGUST, 2025**

SIMILARITY INDEX REPORT



Page 1 of 141 - Cover Page

Submission ID: trn:oid::3618:113596807

Mochamad Wicaksono

Smart Manufacturing Execution System Revolutionizing Industry with IoT and AI (Module Full Stack Website)

 Thesis Computing

Document Details

Submission ID

trn:oid::3618:113596807

130 Pages

Submission Date

Sep 22, 2025, 11:33 AM GMT+7

12,769 Words

Download Date

Sep 22, 2025, 12:18 PM GMT+7

75,491 Characters

File Name

Smart Manufacturing Execution System Revolutionizing Industry with IoT and AI (Module Full St...pdf

File Size

5.0 MB



Page 1 of 141 - Cover Page

Submission ID: trn:oid::3618:113596807

15% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text

Match Groups

- 199 Not Cited or Quoted 14%**
Matches with neither in-text citation nor quotation marks
- 5 Missing Quotations 0%**
Matches that are still very similar to source material
- 0 Missing Citation 0%**
Matches that have quotation marks, but no in-text citation
- 0 Cited and Quoted 0%**
Matches with in-text citation present, but no quotation marks

Top Sources

- 2% Internet sources**
- 1% Publications**
- 14% Submitted works (Student Papers)**

TURNITIN AI REPORT



Page 1 of 132 - Cover Page

Submission ID: trnoid::3618:113596807

Mochamad Wicaksono

Smart Manufacturing Execution System Revolutionizing Industry with IoT and AI (Module Full Stack Website)

Thesis Computing

Document Details

Submission ID

trnoid::3618:113596807

Submission Date

Sep 22, 2025, 11:33 AM GMT+7

Download Date

Sep 22, 2025, 12:12 PM GMT+7

File Name

Smart Manufacturing Execution System Revolutionizing Industry with IoT and AI (Module Full St....pdf

File Size

5.0 MB

130 Pages

12,769 Words

75,491 Characters



Page 1 of 132 - Cover Page

Submission ID: trnoid::3618:113596807

*% detected as AI

AI detection includes the possibility of false positives. Although some text in this submission is likely AI-generated, scores below the 20% threshold are not surfaced because they have a higher likelihood of false positives.

Caution: Review required.

It is essential to understand the limitations of AI detection before making decisions about a student's work. We encourage you to learn more about Turnitin's AI detection capabilities before using the tool.

Disclaimer

Our AI writing assessment is designed to help educators identify text that might be prepared by a generative AI tool. Our AI writing assessment may not always be accurate (i.e., our AI models may produce either false positive results or false negative results), so it should not be used as the sole basis for adverse actions against a student. It takes further scrutiny and human judgment in conjunction with an organization's application of its specific academic policies to determine whether any academic misconduct has occurred.

Frequently Asked Questions

How should I interpret Turnitin's AI writing percentage and false positives?

The percentage shown in the AI writing report is the amount of qualifying text within the submission that Turnitin's AI writing detection model determines was either likely AI-generated text from a large-language model or likely AI-generated text that was likely revised using an AI paraphrase tool or word spinner.

False positives (incorrectly flagging human-written text as AI-generated) are a possibility in AI models.

AI detection scores under 20%, which we do not surface in new reports, have a higher likelihood of false positives. To reduce the likelihood of misinterpretation, no score or highlights are attributed and are indicated with an asterisk in the report (*%).

The AI writing percentage should not be the sole basis to determine whether misconduct has occurred. The reviewer/instructor should use the percentage as a means to start a formative conversation with their student and/or use it to examine the submitted assignment in accordance with their school's policies.

What does 'qualifying text' mean?

Our model only processes qualifying text in the form of long-form writing. Long-form writing means individual sentences contained in paragraphs that make up a longer piece of written work, such as an essay, a dissertation, or an article, etc. Qualifying text that has been determined to be likely AI-generated will be highlighted in cyan in the submission, and likely AI-generated and then likely AI-paraphrased will be highlighted purple.

Non-qualifying text, such as bullet points, annotated bibliographies, etc., will not be processed and can create disparity between the submission highlights and the percentage shown.



ABSTRACT

Easy access to information plays a crucial role in daily life, especially in industrial and manufacturing environments. In this area, engineers rely on a machine's data because it provides them with information about behavior, status changes, or detected anomalies, and allows them to take immediate action. When this information is not readily available—especially if engineers have to manually check the machine or frequently monitor the computer in the office to check if there's an anomaly—it becomes time-consuming and inefficient, particularly if they are away from the computer. These delays can negatively affect productivity and overall efficiency. Therefore, having a system that enables quick access to real-time machine data and alert retrieval is essential to support engineers' decision-making and improve response time, especially in critical situations.

Building a MES can be one of the solutions. Utilizing a web-based application allows the user to easily get information about a machine, with automatic scheduling and notifications. Web programming has advanced over time. Utilizing this advancement, creating a solution such as a web-based Manufacturing Execution System (MES) platform can be useful to overcome this challenge. The integration of an automation system into the platform also allows users, particularly engineers and supervisors, to access important machine information and receive timely alerts through a centralized and user-friendly interface.

In this project, a modular MES web application was developed using Flask as the framework. The access control system implemented so that users can only access features permitted by their role—for example, Admins can manage user accounts, while Supervisors and Technicians have restricted access based on responsibilities. To support

quick response in critical situations, the platform includes a real-time alerting mechanism that notifies users of important events such as machine anomalies or value changes. These alerts are delivered through both the web interface and Telegram, with specific notifications sent directly to targeted users based on their roles. This integration enables technicians and supervisors to stay informed and take action even when they are not actively monitoring the system.

The findings of this project highlight how a modular MES platform can serve as a valuable tool for manufacturers who need real-time access to operational data. By automating reporting and integrating notification systems, the platform enhances both visibility and responsiveness, ultimately contributing to increased productivity and operational efficiency in manufacturing environments.

TABLE OF CONTENTS

	Page
DEDICATION.....	ii
ACKNOWLEDGMENTS	iii
LIST OF TABLES	vii
LIST OF FIGURES	viii
 CHAPTER	
I. INTRODUCTION.....	13
1.1 Background of the Project	13
1.2 Problem Formulation	14
1.3 Solution Employed.....	15
II. SYSTEM ANALYSIS.....	16
2.1 Existing and Proposed System.....	16
2.1.1 Existing and Proposed System.....	16
2.1.2 Proposed Business Process	17
2.2 Global Description of the Product	18
2.2.1 Main Functionality.....	18
2.2.2 User Characteristics	19
2.2.3 Constraints	22
2.2.4 Product Operational Environment	24
2.3 Material Non-Functional Requirements	25
2.4 Specification Testing	25
III. DESIGN	30
3.1 Rational / Systematic Design.....	30
3.1.1 Comparison Tables	30
3.1.2 Quantitative Solution Selection	32
3.1.3 Solution Selection	34
3.2 Hierarchical/Iterative Design.....	34
3.2.1 0-Level Data Flow Diagram	34
3.2.2 1-Level Data Flow Diagram	35

3.2.2.1	1-Level DFD User Management	35
3.2.2.2	1-Level DFD Authentication	35
3.2.2.3	1-Level DFD Password Management	35
3.2.2.4	1-Level DFD End-of-Shift Reporting	36
3.2.2.5	1-Level DFD Notifications	36
3.2.2.6	1-Level DFD Chatbot	36
3.2.2.7	1-Level DFD Profile Management	36
3.2.3	2-Level Data Flow Diagram	37
3.2.3.1	2-Level DFD User Management	37
3.2.3.2	2-Level DFD Authentication	37
3.2.3.3	2-Level DFD Password Management	37
3.2.3.4	2-Level DFD End-of-Shift Reporting	38
3.2.3.5	2-Level DFD Notifications	38
3.2.3.6	2-Level DFD Profile Management	39
3.2.4	Hardware Configuration	39
3.2.5	Software Engineering Design	39
3.2.6	Components Referenced and Library Used	41
3.2.7	Modelling Chart	44
3.2.7.1	Use Case Diagram	44
3.2.7.2	ERD Diagram	45
3.2.7.3	Data Dictionary	45
3.2.7.4	Activity Diagram	52
IV.	IMPLEMENTATION	53
4.1	Design Implementation	53
4.1.1	Functions/Procedure/Class Implementation	53
4.1.2	User interface implementation	82
4.1.2.1	General User Interface Implementation	82
4.1.2.2	User Interface Implementation with Role-Based Access Control	86
4.2	Manual Guide.....	95
4.2.1	System Build Documentation from The Source (Developer Perspective).....	95
4.2.2	User Guide	100
V.	TESTING.....	105
5.1	Functional Testing	105
5.1.1	Functional Testing Results.....	105
5.1.2	Evidence of Tests	108
5.1.2.1	Summary Report Generation Testing	108
5.1.2.2	New Telegram User Chat ID Automatically Inserted into DB Testing	109

5.1.2.3	Role-Based Access Control Testing	110
5.1.2.4	User Authentication Testing	111
5.1.2.5	Reset Password for a New User Testing	113
5.1.2.6	Verification Code for Password Changing Testing	115
5.1.2.7	Unregistered URL Testing	116
5.2	Testing Other Specifications	117
5.2.1	Non-Functional Specifications Testing Results.....	117
VI.	CONCLUSION	118
	REFERENCES	120

LIST OF TABLES

TABLE	Page
2.1 User Characteristics.....	19
3.1 Comparison of Website Implementation.....	30
3.2 Quantitative Solution for Website Implementation Selection.....	32
3.3 Data Dictionary of Users Table.....	45
3.4 Data Dictionary of Employees Table	46
3.5 Data Dictionary of Verification Code Table	47
3.6 Data Dictionary of Notifications Table	48
3.7 Data Dictionary of End-of-Shift Reporting Logs Table.....	49
4.1 Functions/Procedure/Class Implementation	53
4.2 User Interface Implementation	82
4.3 User Interface with RBAC Implementation	86
4.4 User Guide.....	100
5.1 Functional Testing Results	105
5.2 Non-Functional Specifications Testing Results	117

LIST OF FIGURES

FIGURE	Page
2.1 NFR Diagram	25
3.1 0-Level DFD - Overall System.....	34
3.2 1-Level DFD - User Management.....	35
3.3 1-Level DFD - Authentication.....	35
3.4 1-Level DFD - Password Management	36
3.5 1-Level DFD - End-of-Shift Reporting	36
3.6 1-Level DFD - Notifications	36
3.7 1-Level DFD - Chatbot.....	36
3.8 1-Level DFD - Profile Management.....	36
3.9 2-Level DFD - User Management.....	37
3.10 2-Level DFD - Authentication.....	37
3.11 2-Level DFD - Password Management	37
3.12 2-Level DFD - End-of-Shift Reporting	38
3.13 2-Level DFD - Notifications	38
3.14 2-Level DFD - Profile Management.....	39
3.15 Use Case Diagram	44
3.16 Entity Relationship Diagram (ERD)	45
3.17 Activity Diagram	52

4.1	Code Snippet: User Class	54
4.2	Code Snippet: load_user Function.....	55
4.3	Code Snippet: Default Route Handler	56
4.4	Code Snippet: Error Handler Functions	56
4.5	Code Snippet: Login Handler	57
4.6	Code Snippet: Redirect After Login Handler	58
4.7	Code Snippet: Reset Password for First-Time Login Users.....	59
4.8	Code Snippet: Forgot Password	60
4.9	Code Snippet: Verify Code	61
4.10	Code Snippet: Verification Code Generation.....	62
4.11	Code Snippet: Change Password.....	63
4.12	Code Snippet: Role-Based Access Control	64
4.13	Code Snippet: Landing Page	65
4.14	Code Snippet: Before Request.....	66
4.15	Code Snippet: Get User Data	67
4.16	Code Snippet: Register User	68
4.17	Code Snippet: Update User	69
4.18	Code Snippet: Set Status	70
4.19	Code Snippet: Get Chat ID from Username.....	71
4.20	Code Snippet: Get Log Telegram.....	72
4.21	Code Snippet: End-of-Shift Report Table Generation	73
4.22	Code Snippet: End-of-Shift Report Generation	75

4.23	Code Snippet: End-of-Shift Report Delivery Automation	76
4.24	Code Snippet: Sending Message via Telegram Bot	78
4.25	Code Snippet: Chatbot Page Routing (with RBAC)	79
4.26	Code Snippet: Get Unread Notifications	80
4.27	Code Snippet: Get Notifications	81
4.28	Code Snippet: Get Unsolved Count	82
4.29	User Interface – Login Page	83
4.30	User Interface – Reset Password (First-Time Users)	83
4.31	User Interface – Forgot Password (Form 1)	84
4.32	User Interface – Forgot Password (Form 2)	84
4.33	User Interface – Profile Page.....	85
4.34	User Interface – Profile Change Password (Form 1).....	85
4.35	User Interface – Profile Change Password (Form 2).....	86
4.36	User Interface – Sidebar Menus (Admin)	87
4.37	User Interface – Sidebar Menus (Supervisor)	88
4.38	User Interface – Sidebar Menus (Technician).....	89
4.39	User Interface – Landing Page (Admin)	90
4.40	User Interface – Landing Page (Supervisor)	90
4.41	User Interface – Landing Page (Technician).....	91
4.42	User Interface – Dashboard (Admin)	91
4.43	User Interface – Dashboard (Supervisor)	92

4.44	User Interface – Dashboard (Technician).....	92
4.45	User Interface – Chatbot Page (Admin)	93
4.46	User Interface – Chatbot Page (Supervisor).....	94
4.47	User Interface – Chatbot Page (Technician)	94
5.1	Testing Result: End-of-Shift Reporting (Result 1).....	108
5.2	Testing Result: End-of-Shift Reporting (Result 2).....	109
5.3	Users Table on User Management Page (Before Adding a User).....	109
5.4	Users Table in Database (Before Adding a User)	109
5.5	Users Table on User Management Page (After Adding a User)	110
5.6	Users Table in Database (After Adding a User).....	110
5.7	Testing Step: Role-Based Access Control (Step 1).....	110
5.8	Testing Step: Role-Based Access Control (Step 2).....	111
5.9	Testing Result: Role-Based Access Control.....	111
5.10	Testing Step: User Authentication (Step 1).....	112
5.11	Testing Result: User Authentication	112
5.12	Query Result for Validating User Authentication Test	113
5.13	Testing Step: Reset Password for New User (Step 1)	113
5.14	Query Result: Reset Password for New User (Validation 1)	113
5.15	Testing Result: Reset Password for New User (Result 1).....	114
5.16	Testing Result: Reset Password for New User (Result 2).....	114
5.17	Query Result: Reset Password for New User (Validation 2)	115
5.18	Testing Step: Verification Code for Password Change (Step 1)	115

5.19	Testing Result: Verification Code for Password Change.....	115
5.20	Testing Step: Unregistered URL (Step 1)	116
5.21	Testing Result: Unregistered URL	116