

IOT BASED CONDITION MONITORING OF ROTATING MACHINERY AND PREDICTIVE MAINTENANCE



BACHELOR OF MANUFACTURING ENGINEERING TECHNOLOGY WITH HONOUR

2022



Faculty of Mechanical and Manufacturing Engineering Technology



Bachelor of Manufacturing Engineering Technology with Honours

2022

IOT BASED CONDITION MONITORING OF ROTATING MACHINERY AND PREDICTIVE MAINTENANCE

MOHAMAD NAZMEER BIN NAZIR ALI



Faculty of Mechanical and Manufacturing Engineering Technology

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2022

DECLARATION

I declare that this project entitled "IoT Based Condition Monitoring of Rotating Machinery and Predictive Maintenance" is the result of my own research except as cited in the references. Therefore, the Choose item has not been accepted for any degree and is not concurrently submitted in the candidature of any other degree.



APPROVAL

I hereby declare that I have checked this thesis and in my opinion, this thesis is adequate in terms of scope and quality for the award of the Bachelor of Manufacturing Engineering Technology with Honours.

Signature	Hadyan Hafizh Dr. Hadyan Hafizh Pensyarah Kanan Jabatan Teknologi Kejuruteraan Pembuatan Fakulti Teknologi Kejuruteraan Mekanikal dan Pembuatan
Supervisor N	Jame : Dr.Hadyan Hafizh
Date	
	اونيۆم,سيتي تيڪنيڪل مليسيا ملاك
1	INIVERSITI TEKNIKAL MALAYSIA MELAKA

DEDICATION

The research is entirely dedicated to my beloved family, which have inspired me and given me strength when I am exhausted and on the verge of surrendering. In addition, they continue to provide various support in terms of spiritual, moral ,emotional, and financial support.

My appreciated supervisor and panel examiner have advised and encouraged me to complete this study. Additionally, all UTeM lecturers gave me the strength and faith to conquer all obstacles.

Finally, I would like to dedicate this book to the Almighty God for guidance, strength, mental power, protection, inspiration, and long and healthy life.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

ABSTRACT

OEE is Overall Equipment Effectiveness, a hierarchy that measures the performance of a machine for enhanced productivity. The evolution of industry 4.0 is providing immense possibilities to monitor factory equipment like never before. OEE is a powerful tool that helps perform diagnostics as well as manages the production units in different industries. Internet of Things (IoT) technology is helping manufacturing agencies to improve their OEE evaluation with a detailed understanding of equipment performance through instrumentation and analytics. Therefore, the current project proposes an IoT platform for real-time monitoring of factory equipment, such as rotating machinery. The accelerometer sensor inside an Android-based smartphone collects vibration data from rotating machinery. The data is then published to the public MQTT broker via the smartphone's developed application. The application was created with MIT App Inventor, an open-source developer tool. The published data is then subscribed to through Node-RED and visualised in a series of dashboards. A real-time state of the rotating machinery is realised using the proposed system. Furthermore, the time domain data are transformed into the frequency domain in order to validate the collected data from the developed application. The frequencies from the machine are compared to those calculated from the frequency domain using the Fast Fourier Transform (FFT) method. The results showed that both frequencies were in good agreement, proving that the developed application was capable of sensing the correct data. The current project also includes the development of an early warning system as part of a predictive maintenance framework. The system is built with Node-RED as an IoT platform to notify the user of the machine's status via email. The results demonstrated that the developed early warning system could notify the user when the trigger condition is met. Finally, when combined with predictive maintenance. IoT technology has the potential to detect equipment failures in advance. With the introduction of Industry 4.0 in the manufacturing sector, facilities are eager to use IoT technology to gain better insights into operations.

ABSTRAK

OEE ialah Keberkesanan Peralatan Keseluruhan, hierarki yang mengukur prestasi mesin untuk produktiviti yang dipertingkatkan. Evolusi industri 4.0 menyediakan kemungkinan besar untuk memantau peralatan kilang tidak seperti sebelum ini. OEE ialah alat berkuasa yang membantu melaksanakan diagnostik serta mengurus unit pengeluaran dalam industri yang berbeza. Teknologi Internet of Things (IoT) membantu agensi pembuatan mempertingkatkan penilaian OEE mereka dengan pemahaman terperinci tentang prestasi peralatan melalui instrumentasi dan analitik. Oleh itu, projek semasa mencadangkan platform IoT untuk pemantauan masa nyata peralatan kilang, seperti mesin berputar. Sensor pecutan di dalam telefon pintar berasaskan Android mengumpul data getaran daripada mesin berputar. Data tersebut kemudiannya diterbitkan kepada broker MQTT awam melalui aplikasi yang dibangunkan telefon pintar itu. Aplikasi ini dicipta dengan MIT App Inventor, alat pembangun sumber terbuka. Data yang diterbitkan kemudiannya dilanggan melalui Node-RED dan digambarkan dalam satu siri papan pemuka. Keadaan masa nyata jentera berputar direalisasikan menggunakan sistem yang dicadangkan. Tambahan pula, data domain masa diubah menjadi domain frekuensi untuk mengesahkan data yang dikumpul daripada aplikasi yang dibangunkan. Frekuensi dari mesin dibandingkan dengan yang dikira dari domain frekuensi menggunakan kaedah Fast Fourier Transform (FFT). Keputusan menunjukkan bahawa kedua-dua frekuensi adalah dalam persetujuan yang baik, membuktikan bahawa aplikasi yang dibangunkan mampu mengesan data yang betul. Projek semasa juga termasuk pembangunan sistem amaran awal sebagai sebahagian daripada rangka kerja penyelenggaraan ramalan. Sistem ini dibina dengan Node-RED sebagai platform IoT untuk memberitahu pengguna status mesin melalui e-mel. Keputusan menunjukkan bahawa sistem amaran awal yang dibangunkan boleh memberitahu pengguna apabila syarat pencetus dipenuhi. Akhir sekali, apabila digabungkan dengan penyelenggaraan ramalan, teknologi IoT berpotensi untuk mengesan kegagalan peralatan terlebih dahulu. Dengan pengenalan Industri 4.0 dalam sektor pembuatan, kemudahan tidak sabar-sabar untuk menggunakan teknologi IoT untuk mendapatkan pandangan yang lebih baik tentang operasi.

ACKNOWLEDGEMENTS

In the Name of Allah, the Most Gracious, the Most Merciful

With the grace and permission of ALLAH s.w.t. and our prayers to the Prophet Muhammad S.A.W., I was able to do this research. I am indebted to God for providing me with the fortitude, patience, and simplicity necessary to conduct this research.

My heartfelt appreciation to Dr. Hadyan Hafizh, my supervisor, for his encouragement, support, and dedication throughout this study process. I am indebted to him for the information he has imparted to me in exchange for his assistance. Without him, this work would be neither significant nor particularly applicable.

Furthermore, I take this opportunity to thank my family members because of their spiritual and financial supports. They never fail to remind me how I have to focus and keep going despite any difficulties which I have to face. My warmest thanks to my friends, along with my college classmates. Without them, I could honestly would not have completed this. All the long hours are winding through our Final Year Project, and discussing what we know and making decisions about all the proper structure and technicalities helped me produce the best form of work despite my indecisiveness. It was a great learning experience.

Finally, I would like to thanks all the people who have helped in this research process, individually or collectively. There is a genuine appreciation for both the cooperation given

TABLE OF CONTENTS

	PAGE
DECLARATION	
APPROVAL	
DEDICATION	
ABSTRACT	i
ABSTRAK	ii
ACKNOWLEDGEMENTS	iii
CHAPTER 1 INTRODUCTION	1
1.1 Research Background and Motivation	1
1.2 Problem Statement	6
1.3 Research Question	8
1.4 Research Objective	9
1.5 Research Scope and Limitations	10
1.6 Research Design and Planning	11
1.6.1 General Methodology	11
1.6.2 Gantt Chart	13
زیر سیتی بیکنیکل ملیسیا مار Summary	15 اوبير
CHAPTER 2 LITERATURE REVIEW	
2.1 Introduction ERSITI TEKNIKAL MALAYSIA MELA	AKA 16
2.2 IoT Technology	17
2.2.1 Layers in IoT Application	17
2.2.1.1 OSI seven-layer Model	18
2.2.1.2 Five-layer IoT architecture	26
2.2.1.3 Four-layer IoT architecture	28
2.2.1.4 Three-layer IoT architecture	35
2.2.2 Sensor	37
2.2.3 MQTT Protocol	43
2.3 IoT-Based Condition Monitoring and Fault Detection.	49
2.3.1 Recent works in IoT-Based Condition Monitoring.	49
2.3.2 Vibration-Based Fault Detection	53
2.4 Summary	56
CHAPTER 3 METHODOLOGY	57
3.1 Introduction	57
3.2 Research Design and Method	58
3.3 Design of Experiment	61
3.3.1 Application Building and Development	61

3.3.2 Data Collection	63
3.3.3 Data Validation	66
3.3.4 Real-time Warning System	68
3.4 Summary	70
CHAPTER 4 RESULTS AND DISCUSSION	71
4.1 Introduction	71
4.2 Application Building and Development	72
4.2.1 Application Design and Architecture	72
4.2.1.1 Introduction	72
4.2.1.2 The application interface	74
4.2.1.3 Open menu button	76
4.2.1.4 Close menu button	78
4.2.1.5 Accelerometer button	79
4.2.1.6 MQTT Connection button	85
4.2.1.7 Graph Button	92
4.2.1.8 About button	96
4.2.1.9 Exit button	97
4.2.2 MQTT Data Publish and Subscribe	98
4.2.2.1 HiveMQ MQTT Platform Background Introduction	98
4.2.2.2 Reason and Advantages of Using HiveMQ	100
4.2.2.3 Using Public Broker of HiveMQ Platform	104
4.3 Data collection	107
4.3.1 Introduction to Vibration analysis	107
4.3.2 Vibration Analysis Methodology	109
4.3.3 Categories of Vibration Measurement	114
4.3.4 Vibration Analysis Measurement Parameters	117
4.3.5 Benefits of Continuous Vibration Monitoring	119
4.3.6 Measurement Procedure of Vibration Analysis	121
4.4 Measurement of Data	127
4.4.1 Sampling rate calculation	127
4.4.2 Plotting the graph	128
4.5 Graphical result	129
Graphs can be used to visualise enormous amounts of numerical data and to	o depict the
relationships between distinct variables' numerical values. Additionation	ally, they can be
utilised to establish quantitative correlations between variables. A g	raph is a
graphic consisting of one or more points, lines, line segments, curve	s, or regions
that illustrates the variation of one or more variables in contrast to the	ne variation of
one or more other variables.	129
4.5.1 Balanced 10 hertz	130
4.5.1.1 Balanced 10 hertz set 1	130
4.5.1.2 Balanced 10 hertz set 2	131
4.5.1.3 Balanced 10 hertz set 3	132
4.5.2 Unbalanced 10 hertz	133
4.5.2.1 Unbalanced 10 hertz set 1	133
4.5.2.2 Unbalanced 10 hertz set 2	134
4.5.2.3 Unbalanced 10 hertz set 3	135

APPENDICES	179
REFERENCES	174
5.3 Project Potential	172
5.2 Recommendation	171
5.1 Conclusion	169
CHAPTER 5 CONCLUSION AND RECOMMENDATIONS	169
	100
4.8 Summary	168
4.7.2 Real-Time Warning Result	167
4.7.1.2 realures of mode-RED 4.7.1.3 Node-RED setup	159
4.7.1.2 Eastures of Node PED	158
4.7.1 Node-KED and Email Setup	158
4.7 Keal-Time Warning System	158
4.6.4.3 Percentage Error calculation for Case 3: 20 hertz	157
4.6.4.2 Percentage Error calculation for Case 2: 15 hertz	157
4.6.4.1 Percentage error calculation for Case 1: 10 hertz	157
4.6.4 The Percentage of Error Calculation	156
4.6.3.3 The result for case 3: 20 Hertz	154
4.6.3.2 The result for case 2: 15 Hertz	153
4.6.3.1 The result for case 1: 10 Hertz	152
4.6.3 The result of FFT from MATLAB	152
4.6.2 Data validation using MATLAB	150
4.6.1 Introduction to data validation	148
4.6 Data Validation	148
4.5.6.3 Unbalanced 20 hertz set 3	147
4.5.6.2 Unbalanced 20 hertz set 2	146
4.5.6.1 Unbalanced 20 hertz set 1	145
4.5.6 Unbalanced 20 hertz	145
4.5.5.3 Balanced 20 hertz set 3	144
4.5.5.2 Balanced 20 hertz set 2	142
4.5.5 Dataliced 20 hertz set 1	142
4.5.4.5 Unbalanced 15 hertz set 5	141
4.5.4.2 Unbalanced 15 hertz set 2	140
4.5.4.1 Unbalanced 15 hertz set 1	139
4.5.4 Unbalanced 15 hertz	139
4.5.3.3 Balanced 15 hertz set 3	138
4.5.3.2 Balanced 15 hertz set 2	137
4.5.3.1 Balanced 15 hertz set 1	136
4.5.3 Balanced 15 hertz	136

LIST OF TABLES

TABLE	TITLE	PAGE
Table 2.1 Groups of	IoT by their functions	29
Table 2.2 Networkin	g Technologies	30
Table 2.3 Protocol ir	ı IoT ecosystem	32
Table 2.4 Terms use	d in MQTT	47
Table 4.1 Button nar	ne and function on the home screen of the Application	75
Table 4.2 Accelerom	eter sensor and MQTT setup	80
Table 4.3 Public MQ	TT setup	86
Table 4.4 Button fur	nction on a graph section	93
Table 4.5 Benefits of	f the HiveMQ MQTT platform	101
Table 4.6 Step to use	HiveMQ broker platform	105
Table 4.7 Method of	vibration analysis	108
Table 4.8 Four basic	principles of vibration analysis AYSIA MELAKA	110
Table 4.9 List of form	ms of analysis, calculations and algorithms used to determine	ne
different	aspects of vibration analysis	112
Table 4.10 Categorie	es of vibration measurement and its explanation	114
Table 4.11 Procedure	e of taking data using SENSE IT	121
Table 4.12 Domain of	characteristic of node-RED	159
Table 4.13 Nodes de	tail that has been used in this project	162

LIST OF FIGURES

FIGURE TITLE	PAGE
Figure 1.1 Flow Chart General Methodology	11
Figure 1.2 Gantt chart of Project semester 1	13
Figure 1.3 Gantt chart of Project semester 2	13
Figure 2.1 Architecture of IoT of three-layer, four-layers and five-layers	17
Figure 2.2 Basic seven-layers OSI Model	18
Figure 2.3 Seven-layers Open Systems Interconnection Model	19
Figure 2.4 Physical layer representation	20
Figure 2.5 Data Link layer representation	20
Figure 2.6 Network layer representation	21
Figure 2.7 Transport layer representation	22
Figure 2.8 Session layer representation	23
Figure 2.9 Presentation layer representation	24
Figure 2.10 Application layer representation	25
Figure 2.11 Five-layers IoT architecture	26
Figure 2.12 Four-layers IoT architecture	28
Figure 2.13 Four-layers IoT summary	29
Figure 2.14 Common architecture of IoT	33
Figure 2.15 Three-layer IoT architecture	35
Figure 2.16 Three dimensional axes of accelerometer sensor	37
Figure 2.17 Diagram of MEMS Capacitive Accelerometer	39

Figure 2.18 LIS302DL accelerometer sensor	41
Figure 2.19 Simple Wi-Fi Configuration	44
Figure 2.20 MQTT workflow	45
Figure 2.21 Topology Diagram of IoT-based Condition Monitoring	50
Figure 2.22 Usage of a vibration sensor	51
Figure 3.1 Research Methodology	58
Figure 3.2 Work Flow of SENSE IT application developed	61
Figure 3.3 Full flow of SENSE IT application	62
Figure 3.4 Data Collection Framework	63
Figure 3.5 Flow chart for the data collection process	65
Figure 3.6 Data Validation Framework	66
Figure 3.7 Process of data validation	67
Figure 3.8 Real-time alert system	68
Figure 3.9 Work flow for warning system alert	69
Figure 4.1 Logo of the SENSE IT Application	72
Figure 4.2 Home page of MIT application inventor	73
Figure 4.3 Code-blocks on the application inventor	73
Figure 4.4 Logo of SENSE IT on smartphones	74
Figure 4.5 Home screen of SENSE IT Application	74
Figure 4.6 Side menu screen with IoT, PdM and AI notes	76
Figure 4.7 Interface of IoT and PdM info	77
Figure 4.8 Function of the close menu	78
Figure 4.9 Accelerometer reading and the MQTT setup	79
Figure 4.10 Multiple platform connection option	85
1A	

Figure 4.11 Interface of the graph section	92
Figure 4.12 HTML code used to generate-graph	94
Figure 4.13 Interface of the about section, which explains the SENSE IT application	96
Figure 4.14 Exit button image	97
Figure 4.15 Example of devices and enterprises that connect to the HiveMQ MQTT	
platform	98
Figure 4.16 Several key organisations that joined the HiveMQ community	99
Figure 4.17 MQTT browser client interface	104
Figure 4.18 UrsAI2MQTT extension where broker, password and username are	
setups	105
Figure 4.19 Example graph of time waveform analysis	109
Figure 4.20 Example graph of frequency-domain vibration signal	111
Figure 4.21 Graph of acceleration, displacement and velocity	118
Figure 4.22 Snapshot figure of data analysis to plot the graph	128
Figure 4.23 Balanced 10 hertz set 1 for X, Y, and Z-Axis	130
Figure 4.24 Balanced 10 hertz set 2 for X, Y, and Z-Axis	131
Figure 4.25 Balanced 10 hertz set 3 for X, Y, and Z-Axis	132
Figure 4.26 Unbalanced 10 hertz set 1 for X, Y, and Z-Axis	133
Figure 4.27 Unbalanced 10 hertz set 2 for X, Y, and Z-Axis	134
Figure 4.28 Unbalanced 10 hertz set 3 for X, Y, and Z-Axis	135
Figure 4.29 Balanced 15 hertz set 1 for X, Y, and Z-Axis	136
Figure 4.30 Balanced 15 hertz set 2 for X, Y, and Z-Axis	137
Figure 4.31 Balanced 15 hertz set 3 for X, Y, and Z-Axis	138
Figure 4.32 Unbalanced 15 hertz set 1 for X, Y, and Z-Axis	139

Figure 4.33 Unbalanced 15 hertz set 2 for X, Y, and Z-Axis	140
Figure 4.34 Unbalanced 15 hertz set 3 for X, Y, and Z-Axis	141
Figure 4.35 Balanced 20 hertz set 1 for X, Y, and Z-Axis	142
Figure 4.36 Balanced 20 hertz set 2 for X, Y, and Z-Axis	143
Figure 4.37 Balanced 20 hertz set 3 for X, Y, and Z-Axis	144
Figure 4.38 Unbalanced 20 hertz set 1 for X, Y, and Z-Axis	145
Figure 4.39 Unbalanced 20 hertz set 2 for X, Y, and Z-Axis	146
Figure 4.40 Unbalanced 20 hertz set 3 for X, Y, and Z-Axis	147
Figure 4.41 Snapshot of code used in MATLAB to filter and transform the data.	150
Figure 4.42 Time-domain to frequency-domain transform using FFT	150
Figure 4.43 Raw data for 10-hertz frequency	152
Figure 4.44 Filtered data for 10-hertz frequency	152
Figure 4.45 Raw data for 15-hertz frequency	153
Figure 4.46 Filtered data for 15 hertz frequency	153
Figure 4.47 Raw data for 20-hertz frequency	154
Figure 4.48 Filtered data for 20-hertz frequency	154
Figure 4.49 Percentage error formula	157
Figure 4.50 Key features of Node-RED	160
Figure 4.51 Interface of the Node-RED environment on a palette with data displayed	
on the side	161
Figure 4.52 Custom setting for a specific function, such as saving the file as CSV	161
Figure 4.53 Data displayed by debug node with timestamps such as date and time	164
Figure 4.54 Data saved as a CSV file with a timestamp	165
Figure 4.55 SENSE IT dashboard that display live data on gauge and line chart graph	165

rigure 4.50 Live data of acceleration in meter per second square on a gauge	00
Figure 4.57 Live data of acceleration in meter per second square on a line chart 16	66
Figure 4.58 Email alert and the content of the email received each time abnormalities	

occur 167



LIST OF SYMBOLS AND ABBREVIATIONS

IoT	-	Internet of Things
MIT	-	Massachusetts Institute of Technology
App	-	Application
PdM	-	Predictive Maintenance
MQTT	-	Message Queuing Telemetry Trasport
Wi-Fi	-	Wireless Fidelity
RxM	-	Prescriptive maintainance
SOP	-	Standard Operating Procedure



LIST OF APPENDICES

APPENDIX	TITLE	PAGE
APPENDIX A The	e environment of coding in MIT Application	179
APPENDIX B Coo	le block in MIT Application	180
APPENDIX C Coo	le block in UrsAIMQTT Extension	181
APPENDIX D Inte	erface of MIT developed Application	182
APPENDIX E Hiv	eMQ cloud server receiving data in real-time	183
APPENDIX F Noc	le-RED collecting data in real-time	184
APPENDIX G SE	NSE IT dashboard for gauge meter of acceleromter	185
APPENDIX H Cer	tificate of completion from MIT APP inventor	186
APPENDIX I Cert	ificate of achievement from MIT APP inventor	187
APPENDIX J Bad	ges and recognition from MIT APP inventor	188
APPENDIX K Cer	tificate of completion from node-RED	189
APPENDIX L Bac	اونیوم سینی ترIges and recognition from node-RED	190
APPENDIX M Ce	rtificate from International MAIoT Challenge 2021	191

CHAPTER 1

INTRODUCTION

1.1 Research Background and Motivation

The advent regarding Industry 4.0 offers medium and small-sized enterprises (SMEs) an opportunity for improved production processes. Considering that such businesses have restricted investment ability, innovations such as algorithms based on artificial intelligence, the Internet of Things (IoT), and cloud computing, which are typical for industry 4.0, will benefit from their low-cost features (Dias et al., 2021). However, there are still challenges to research, counting also about central background of Industries 4.0, such as interoperability, modularity, distributed data processing in large quantities generated by these disparate devices, and their interaction with other industrial processing systems (Souza et al., 2021).

The management model of the Internet of Things (IoT) is the management process and a scientific management model that effectively connects perception and communication. Next, the successful integration of the cloud and application fosters technical innovation, managerial innovation, and system innovation (Wu & Xiao, 2021). The emergence of connected devices and instrumented settings, together with existing data analysis and cognition capabilities, created the circumstances for the Internet of Things to be consolidated (da Silva et al., 2016). Linking "things" on the Internet has become commonplace since the Internet's inception in 1989. The Trojan Room coffee pot is most likely the first of its kind. (Gupta et al., 2010). John Romkey invented the very first Internet in 1990, coining the term 'the computer,' a toaster that can be turned on and off over the Internet. Meanwhile, in 1994, Steve Mann discovered WearCam, and It worked almost in real-time on a 64-bit processor machine. Moving on, Paul Saffo gave the first summary of the sensors and their potential course of action in 1997. Moving next, Kevin Ashton, who positioned as an executive director at Auto-ID centre of MIT, developed the term "Internet of Things" in 1999. They also developed a global RFID-based object detection program in the same year (Ashton, 2009). such as significant growth in promoting the Internet of Things; they announced in 2000 that they would make an 'intelligent' refrigerator that checks if the food in it is running low on shelf life, replenishing itself, and informs you (Dvali & Belonin, 2009).

The internet of things is the collaborative network of physical devices with APIs which can write programs capable of sensing, connecting, and navigating to a chosen internet domain. Such objects are embedded in electronics (microcontrollers and transceivers), applications, cameras, actuators, and network connections to encapsulate data for specific protocols (Atzori et al., 2010). Thus, IoT connects with others in new ways to computers, systems, and facilities beyond Machine-to-Machine (M2M) communication, and it has a broad application in numerous fields. (N. Sharma et al., 2019).

The Internet of Things (IoT) is a central pillar that is a fundamental component of Predictive Maintenance or, in short, PdM (Kwon et al., 2016). It allows machine behaviour to be transformed from computers into digital signals, which PdM then uses. The IoT technology can continuously stream numerous sensor data such as temperature, Vibration, and many more. This is for real-time monitoring purposes and from other outlets, such as the Programmable Logic Controller (PLC), the Computerized Maintenance Management System (MES) terminals, possibly also an ERP (Enterprise Resource Management System). Such understanding constitutes the base for determining approaches to PdM (Compare et al., 2019). Currently, for actuators, sensors, and appropriate data systems, alongside developed devices' performance conditions, can monitor the installation and remote control of automated industrial systems. Besides that, due to the possibility of numerous sensors generating big data. This scenario has the potential to generate massive volumes of data throughout operation (Manyika et al.,2011). Next, models, analytic can test the related data, optimise industrial procedures of operation, and stretch the observed machines' life period. All of these are the primary objectives and explain why the industrial revolution wave is gaining momentum (da Silva et al., 2016).

IoT products have become viral and have numerous demands. Most people prefer to use them because it is easy to use intelligent IoT products in home appliances such as light, fans, air conditioning, and even certain kitchen appliances like coffee machines (Gubbi et al., 2013). There are issues in handling the collection of IoT devices in a home environment. For instance, the machines may be in several places in a house, and there may not be a way to handle all the appliances in one home. Moreover, there is no way to determine whether a particular sensor or another IoT device module is about to get corrupted or permanently damaged. It is an effort to incorporate an architecture at the system level that uses the computational capacity of the Machine Learning algorithms to predict a compromised sensor or module and, besides, will be able to notify the user of the issue in such a manner that the user can easily avoid it from happening, saving not only money or other devices from affected, but also users themselves from possible physical harm when a machine malfunction.

Though it is easy to think that devices can be substituted out if they get permanently damaged, there is more to this issue. The challenges can range from IoT devices permanently damaged to inflicting severe damage on other devices that data may be transmitted to perform tasks. Besides that, it may inflict life-threatening harm to home environment

customers when the system is tightly linked to handling users' everyday lives. Furthermore, the challenges do not just result in business collapse, and they could end up physically destroying them (Adikari, 2017).

Rotating machines are widely utilised in the industry (from minor to significant businesses) and are often driven by electric motors. Moreover, continuous track related to these machines is crucial so that unforeseen output stops due to malfunctions or failures are able to hinder. An intelligent faulty diagnostic system is therefore used for identifying and recognising anomalies all the while computer activity. These are known as condition monitoring systems, and they are capable of detecting mistakes or deteriorating processes in advance of a functional failure (Vanraj et al., 2016). Manufacturing processes are fundamental to industry, and among the several types of equipment, one that is ubiquitous are electrical's motors. The circumstance simply explains that electrical power consumption is attributed to either motors or engines by 40 per cent. This was accounting for the electrical power consumed by the industry for two over three of total consumed (Saidur, 2010). Although the proposed infrastructure is sufficiently versatile to be used in many industrial scenarios, however, in a demonstration system to follow the operation of the sensor and the actuator controlled system in the context of exhaust monitoring and feedback management cloud platform for an industrial electric motor (da Silva et al., 2016).

Predictive Maintenance or known as PdM, is a technique for monitoring machinery status to avoid costly failure and also to execute Maintenance only when essential. PdM has evolved from the earliest methodology that is a visual inspection, to automated systems utilising modern techniques, that is, signal processing. Historically, for Maintenance has created a trade-off situation in where it is an option between maximising a part's useful life at the expense of machine downtime, referred to as run to failure, and maximising uptime by replacement of potentially good parts earlier, referred to as time-based PdM, which has

been shown to be ineffective for the overwhelming majority of equipment components, which are now considered as unreliable and flaws (Mobley, 2002). Basically, Predictive Maintenance proposes to eliminate hurdles by enabling businesses to forecast Maintenance more accurately to control it. It optimises the life cycle of assets and reduces preventative maintenance activities. With substantial time and cost savings, there are substantial efficiency and reliability improvements (Traini et al., 2019).

The long-term cost of manufacturing is vital to all operating plants. Keeping an eye on the environment is essential to the running of the programme. Many predictive maintenance programmes use Vibration. That, however, means that the typical monitoring technique relies heavily on human factors and is therefore subject to inaccuracy. Thus, to solve these problems, It can be formulated the following research question and objective

As indicated by Divorski points out on how to improve while assessing data quality, one should first examine the quality of available data and consider data restrictions (Divorski & Scheirer, 2001). Furthermore, according to Nakajima, overall equipment effectiveness or known as OEE, is a critical metric under the TPM paradigm. While practitioners have traditionally used OEE as an operational metric to track production performance, it can also be used as a proxy for process improvement initiatives in a manufacturing setting (Sohal et al.,2011). However, this data could be collected in real-time more efficiently using IoT techniques.

1.2 Problem Statement

Predictive maintenance (PdM) is a type of maintenance programme based on realtime requirements. Rather than scheduling maintenance activities based on average industrial or in-house statistics, PdM employs direct monitoring of operating conditions, efficiency, heat distribution, vibration, and/or noise to ascertain the real mean-time-to-failure or efficiency loss that would be detrimental to plant operations for all essential systems in the plan. The installation of a thorough predictive maintenance programme enables and provides accurate information on the actual operational conditions for essential equity, including their effectiveness, just as the real mechanical condition of each machine train and interaction framework. This information provides maintenance management with the accurate information necessary to plan and schedule maintenance activities effectively (Bousdekis et al., 2019).

Each maintenance team is aware that failing to manage assets properly can result in frequent and lengthy failures, a high volume of unplanned labour, and a drop in productivity. However, the costs associated with unexpected corrective Maintenance are less obvious—overtime wages, accelerated prices, and even safety issues. This type of Maintenance might have a detrimental effect on the asset's usable life. Additionally, excessive asset upkeep might be wasteful (Selcuk, 2017). Moreover, excessive maintenance expenditures waste valuable resources, which has a direct impact on cost and profitability. Nevertheless, technology has grown to supplement human intelligence and provide more precise monitoring of asset health. However, there is a gap in assessing the maintenance data; there is no access to real-time machine health data. The manager must be well-informed regarding the machine health data in making critical decisions. Therefore, the current project offers a solution in providing real-time visualisation of machine health data. Hence, the manager can

mitigate the risk and improve decision-making over time. The expected outputs from the current research projects include IoT devices and architecture for monitoring of machine health data as a tool to support decision making (Wieger et al., 2020).



1.3 Research Question

The following summarises the research question for this project:

- a) How to design an IoT platform in the context of predictive maintenance for industrial machinery?
- b) How to automatically notify the user regarding the health state of industrial machinery using IoT technology?
- c) What is the practical and industrial applicability of the developed system?



1.4 Research Objective

The research objective of this project is summarised as the following:

- a) To design and develop an IoT (Internet of Things) platform for a machine monitoring system.
- b) To design and develop a warning system for faulty operation detection of a rotating machine.
- c) To demonstrate the applicability of the developed platform for practical cases.



1.5 Research Scope and Limitations

The scopes in this project are based on understanding the limitation and concept of the IoT technology in predictive maintenance for rotating machinery. Besides that, the current project is using the IoT platform with open-source software such as MIT application inventor and public cloud MQTT broker. Furthermore, the sensor deployed in this project is a vibration sensor or known as an accelerometer embedded in an Android-based smartphone. The data obtained from the sensor are analysed by using signal processing techniques such as Fast Fourier Transform(FFT). The selected case study for the current project is limited to which data is collected using a sensor in 1 type of smartphone only.



1.6 Research Design and Planning

1.6.1 General Methodology



Figure 1.1 Flow Chart General Methodology

Figure 1.1 shows the general methodology of the current project. In the figure, two discrete PSM 1 and PSM 2 plans represent two semesters'. PSM 1 main task is creating a base for the project, such as developing the application. PSM 2 is about collecting data, data analysis and demonstrates the hands-on or applicability of this project.

Firstly, in PSM 1, the first task is to define problems and formulate objectives for the research. After that, formulate the design of the experiment, which decides how the experiment will be carried out. Next, the task is to perform a literature review from existing papers regarding related topics. In addition to that, a preliminary data collection process takes place that is data of accelerometer is taken. After all the step is done, the process of report writing takes place for chapter 1,2 and 3. For PSM 2 first task is to collect data and perform simulation on the rotating machine. After that, data analysis is performed alongside with data validation in order to make sure the data obtained is valid and useable. Next is the result obtained from the previous process is discussed and then resume findings and suggestions for future works, such as which area can be improved and what can be modified. Lastly, report writing for chapter 4 and 5 take place.

1.6.2 Gantt Chart



Figure 1.3 Gantt chart of Project semester 2

As shown in Figures 1.2 and 1.3, this PSM project is divided into two parts because it is conducted for two semesters, and Gantt chart activity was represented by date for finished PSM 1 meanwhile represented by week for planned PSM 2. Basically, on PSM 1, there are three main parts of work that need to be completed. 1) formulation of research background, research question, and research objectives, 2) literature review, and 3) designing a research methodology. It was reported as Chapter 1 to 3 in the final report of PSM1. Chapter 1 mainly discussed about the project background, problem statements, and formulation of objectives. Chapter 2 mainly discussed about the literature review, and Chapter 3 described the methods for conducting research. This chapter also includes the introduction of IoT technologies and IoT-based condition monitoring, along with faulty detection based on IoT devices. Also, from the literature review, it can be seen how IoT-based devices are being used in the industry.

Furthermore, Chapter 3 contains the research methodology. This chapter discussed the research design and method applied to the current project. The research design and methods are divided into data collection, data analysis, and data validation. Chapter 4 presents about data collection, analysis, and validation. Lastly is the conclusion and recommendation that present about summary of this project in Chapter 5.

AALAYSI.

A milestone is a discrete moment in the life cycle of a project that is used to track progress toward the project's end goal. Milestones are used in project management to indicate the start and finish dates of a project, external evaluations or input, budget checks, and the submission of a major deliverable, among other things. A milestone is a reference point inside a project that denotes a noteworthy event or a branching decision point. For PSM 2, Four milestones have been achieved that is milestone 1 for connecting data to HiveMQ cloud broker, milestone two for finishing the SENSE IT application, milestone three for finishing node-RED system and lastly, milestone 4 for developing warning system.

1.7 Summary

This chapter explains the study of research background, problems statement, research questions, research objectives, and scope and limitations. The research background and motivation are written based on the identified problems, three research questions and objectives have been formulated. An in-depth study of the literature review is presented in the next chapter.



CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This section includes a review of the available literature on IoT (Internet of Things) technology, which is a new revolution that enables a wide variety of things to connect with one another via wireless transmission technology. In a nutshell, the Internet of Things is a network of interconnected computing devices that includes mechanical and digital machines, objects, animals, and people that have been assigned unique identifiers (UIDs) and the capability to transfer data over a network without the need for human-to-human or human-to-computer interaction. As a result, intelligent procedures and enhanced services are enabled. To begin this chapter, a review of Internet of Things technology will be presented, including sensors and MQTT protocol. Additionally, this chapter will explain about the application of the Internet of Things for condition monitoring alongside with fault detection in rotating machines. Additionally, this chapter will discuss current developments in Vibration-Based Fault Detection.
2.2 IoT Technology

2.2.1 Layers in IoT Application

In its simplest form, IoT architecture is a collection of disparate components: sensors, protocols, actuators, cloud services, and layers. Due to the intricacy of IoT architecture, it is divided into four stages. This figure gradually integrates these disparate components into a complex and coherent network.

Basically, there are three models of layers in IoT architecture, as shown below in Figure 2.1: three-layer, four-layer (the standard IoT layer), and five-layer. Additionally, there is a seven-layer OSI model from which the IoT 4 layer emerges.



Figure 2.1 Architecture of IoT of three-layer, four-layers and five-layers

2.2.1.1 OSI seven-layer Model

The OSI Model, known as Open Systems Interconnection Model, is a conceptual framework for describing the functions of a networking system. The OSI model abstracts computer functions into a standardized set of rules and standards that enables interoperability between various products and software. According to the OSI reference model, communication between computing systems is divided into seven distinct abstraction layers, as shown below in figure 2.2: Physical, Data Link, Network, Transport, Session, Presentation, and Application.



Figure 2.2 Basic seven-layers OSI Model

The OSI was established in 1984 by the International Organization for Standardization when network computing was in its infancy (ISO). While the OSI Model does not always map exactly to real systems, it is nonetheless used to explain network architecture today. To sum up, the OSI Model can be seen as a universal language for computer networking. It's based on the concept of splitting up a communication system into seven abstract layers, each one stacked upon the last. Moving on, the 7 Layers of the OSI Model is made up of Physical Layer, Data Link Layer, Network Layer, Transport Layer, Session Layer, Presentation Layer and Application Layer as shown below in figure 2.3. Each layer of the OSI Model is responsible for a distinct function and interacts with the layers above and below it.



Figure 2.3 Seven-layers Open Systems Interconnection Model

The Physical Layer



Sending cable

Bitstream

Receiving cable

Figure 2.4 Physical layer representation

Firstly, as shown in figure 2.4, the physical layer is the lowest layer of the OSI Model is concerned with electrically or optically transmitting raw unstructured data bits across the network from the physical layer of the sending device to the physical layer of the receiving device. It can include specifications such as voltages, pin layout, cabling, and radio frequencies. At the physical layer, one might find "physical" resources such as network hubs, cabling, repeaters, network adapters or modems.



Figure 2.5 Data Link layer representation

Secondly, Data Link Layer are as shown in figure 2.5, where at the data link layer, directly connected nodes are used to perform node-to-node data transfer where data is packaged into frames. The data link layer also corrects errors that may have occurred at the physical layer. The data link layer encompasses two sub-layers of its own. Media access control (MAC) provides flow control and multiplexing for device transmissions over a

network. The second, the logical link control (LLC), provides flow and error control over the physical medium and identifies line protocols.

The data link layer is very similar to the network layer, except the data link layer facilitates data transfer between two devices on the SAME network. The data link layer takes packets from the network layer and breaks them into smaller pieces called frames. Like the network layer, the data link layer is also responsible for flow control and error control in intra-network communication (The transport layer only does flow control and error control for inter-network communications).



Thirdly, Network Layer, as shown in figure 2.6, is responsible for receiving frames from the data link layer and delivering them to their intended destinations based on the addresses contained inside the frame. The network layer finds the destination by using logical addresses, such as IP (internet protocol). At this layer, routers are a crucial component used to quite literally route information where it needs to go between networks.

The network layer is responsible for facilitating data transfer between two different networks. If the two devices communicating are on the same network, then the network layer is unnecessary. The network layer breaks up segments from the transport layer into smaller units, called packets, on the sender's device and reassembles these packets on the receiving device. The network layer also finds the best physical path for the data to reach its destination, known as routing.



Figure 2.7 Transport layer representation

Next, Transport Layer, as shown in figure 2.7, manages data packets' delivery and error checking. It regulates the size, sequencing, and ultimately the transfer of data between systems and hosts. One of the most common examples of the transport layer is TCP or the Transmission Control Protocol.

Layer 4 is responsible for end-to-end communication between the two devices. This includes taking data from the session layer and breaking it up into chunks called segments before sending it to layer 3. The transport layer on the receiving device is responsible for reassembling the segments into data the session layer can consume.

The transport layer is also responsible for flow control and error control. Flow control determines an optimal transmission speed to ensure that a sender with a fast connection doesn't overwhelm a receiver with a slow connection. The transport layer performs error control on the receiving end by providing that the data received is complete and requesting a retransmission if it isn't

The Session Layer



Figure 2.8 Session layer representation

After that, Session Layer, as shown in figure 2.8, controls the conversations between different computers. A session or connection between machines is set up, managed, and terminated at layer 5. Session layer services also include authentication and reconnections.

This is the layer responsible for opening and closing communication between the two devices. The time between when the communication is opened and closed is known as the session. The session layer ensures that the session stays open long enough to transfer all the data being exchanged and then promptly closes the session in order to avoid wasting resources.

The session layer also synchronizes data transfer with checkpoints. For example, if a 100-megabyte file is being transferred, the session layer could set a checkpoint every 5 megabytes. In the case of a disconnect or a crash after 52 megabytes have been transferred, the session could be resumed from the last checkpoint, meaning only 50 more megabytes of data need to be transferred. Without the checkpoints, the entire transfer would have to begin again from scratch.



Figure 2.9 Presentation layer representation

Also, as shown in figure 2.9, the Presentation Layer formats or translates data for the application layer based on the syntax or semantics that the application accepts. Because of this, it is at times also called the syntax layer. This layer can also handle the encryption and decryption required by the application layer.

This layer is primarily responsible for preparing data so that the application layer can use it; in other words, layer 6 makes the data presentable for applications to consume. The presentation layer is responsible for translation, encryption, and data compression. Two communicating devices communicating may be using different encoding methods, so layer 6 is responsible for translating incoming data into a syntax that the application layer of the receiving device can understand. If the appliances are communicating over an encrypted connection, layer 6 is responsible for adding the encryption on the sender's end as well as decoding the encryption on the receiver's end so that it can present the application layer with unencrypted, readable data.

In addition to that, the presentation layer is also responsible for compressing data it receives from the application layer before delivering it to layer 5. This helps improve the speed and efficiency of communication by minimizing the amount of data that will be transferred.





Figure 2.10 Application layer representation

Lastly, as shown in figure 2.10, the application layer where both the end user and the application layer interact directly with the software application sees network services provided to end-user applications such as a web browser or Office 365. The application layer identifies communication partners, resource availability and synchronizes communication.

This is the only layer that directly interacts with data from the user. Software applications like web browsers and email clients rely on the application layer to initiate communications. But it should be made clear that client software applications are not part of the application layer. Rather the application layer is responsible for the protocols and data manipulation that the software relies on to present meaningful data to the user. Application layer protocols include HTTP as well as SMTP (Simple Mail Transfer Protocol is one of the protocols that enables email communications).

2.2.1.2 Five-layer IoT architecture

The three-layer architecture defines the main idea of the Internet of Things. Still, it is not sufficient for research on IoT because research often focuses on finer aspects of the Internet of Things. That is why we have many more layered architectures. One is the five-layer architecture, which additionally includes the processing and business layers. The five layers, as shown in figure 2.11 are consist of perception, transport, processing, application, and business layers. The role of the perception and application layers is the same as the architecture with three layers.



Figure 2.11 Five-layers IoT architecture

Firstly, the perception layer is the physical layer, which has sensors for sensing and gathering information about the environment. It senses some physical parameters or identifies other smart objects in the environment. This is the physical layer of the architecture. This is where the sensors and connected devices come into play as they gather various amounts of data as per the need of the project. These can be the edge devices,

sensors, and actuators that interact with their environment. Next is the transport layer transfers the sensor data from the perception layer to the processing layer and vice versa through networks such as wireless, 3G, LAN, Bluetooth, RFID, and NFC. After that, the processing layer is also known as the middleware layer. It stores, analyze, and processes huge amounts of data from the transport layer. It can manage and provide various services to the lower layers. It employs many technologies such as databases, cloud computing, and big data processing modules. Next is the application layer, which delivers analytics, reporting, and device control to end-users.

For the application layer, which addresses business requirements at this layer, information is analyzed by software to answer key business questions. Hundreds of IoT applications vary in complexity and function, using different technology stacks and operating systems. Examples include device monitoring and control software, mobile apps for simple interactions, business intelligence services, and analytic solutions using machine learning. Currently, applications can be built right on top of IoT platforms that offer software development infrastructure with ready-to-use instruments for data mining, advanced analytics, and data visualization. Otherwise, IoT applications use APIs to integrate with middleware.

The Business Layer, which implements data-driven solutions, manages the whole IoT system, including applications, business and profit models, and users' privacy. The information generated at the previous layers brings value if only it results in problem-solving solutions and achieving business goals. New data must initiate collaboration between stakeholders, who in turn introduce new processes to enhance productivity. The decisionmaking usually involves more than one person working with more than one software solution. For this reason, the business layer is defined as a separate stage, higher than a single application layer.

2.2.1.3 Four-layer IoT architecture

This model takes a slightly different ordering to the three-layer concept, with an application, data processing, network, and perception or sensor layers. In the four-layer IoT architecture, as shown in figure 2.12, there are perception layers, connectivity or transport layers, processing layers, and application layers.



Firstly, the perception layer will be hosting smart things, and the perception layer UNIVERSITI TEKNIKAL MALAYSIA MELAKA

will convert analog signals into digital data and vice versa. The initial stage of any IoT system embraces a wide range of "things" or endpoint devices that act as a bridge between the real and digital worlds. They vary in form and size, from tiny silicon chips to large vehicles. By their functions, IoT things can be divided into the following large groups as shown in table 2.1 below.

Item	Explanation
Sensors	such as probes, gauges, meters, and others. They collect physical
	parameters like temperature or humidity, turn them into electrical signals,
	and send them to the IoT system. IoT sensors are typically small and
	consume little power
Actuators	translating electrical signals from the IoT system into physical actions.
	Actuators are used in motor controllers, lasers, robotic arms.
Machines	connected to sensors and actuators or having them as integral parts.
and devices	
	WALAYSIA HA

Table 2.1 Groups of IoT by their functions

It's important to note that the architecture puts no restriction on the scope of its components or their location. The edge-side layer can include just a few "things" physically placed in one room or myriads of sensors and devices distributed across the world.



Figure 2.13 Four-layers IoT summary

Secondly, the connectivity or transport layer transferring data from the physical layer to the cloud and vice versa via networks and gateways. The second level is in charge of all communications across devices, networks, and cloud services that make up the IoT infrastructure, as shown in figure 2.13. The connectivity between the physical layer and the cloud is achieved in two ways:

- 1. Directly, using TCP or UDP/IP stack.
- Via gateways hardware or software modules were performing translation between different protocols as well as encryption and decryption of IoT data.

The communications between devices and cloud services or gateways involve different networking technologies. The networking technologies are listed in table 2.2 below:

Thomas		Evelopetion
nem	5	Explanation
	- Z	C A A
Ethernet	H	connects stationary or fixed IoT devices like security and video
	FISTAN	cameras, permanently installed industrial equipment, and gaming
		consoles.
) ملاك	اونيۈم سيتي تيڪنيڪل مليسي
	UNIVER	SITI TEKNIKAL MALAYSIA MELAKA
Wi-Fi		the most popular wireless networking technology is a great fit for
		data-intensive IoT solutions that are easy to recharge and operate
		within a small area. A good example of use is smart home devices
		connected to the electrical grid.
NFC (Ne	ar Field	enables simple and safe data sharing between two devices over a
Communic	cation)	distance of 4 inches (10 cm) or less.

MALAY	Fable 2.2 Net	working Technologies
V	190	

Dhuataath	is widely used by weapphles for short way as communications. To
Bluetooth	is widely used by wearables for short-range communications. To
	meet the needs of low-power IoT devices, the Bluetooth Low-
	Energy (BLE) standard was designed. It transfers only small data
	portions and doesn't work for large files.
LPWAN (Low-	was created specifically for IoT devices. It provides long-range
power Wide-area	wireless connectivity on low power consumption with a battery
Network)	life of 10+ years. Sending data periodically in small portions, the
	technology meets the requirements of smart cities, smart
MAL	buildings, and smart agriculture (field monitoring).
Sec.	
TER	
ZigBee	Zigbee is a low-power wireless network for carrying small data
" anine	packages over short distances. The extraordinary thing about
) ملاك	ZigBee is that it can handle up to 65,000 nodes. Created
UNIVER	specifically for home automation, it also works for low-power
	devices in industrial, scientific, and medical sites.
Cellular networks	offer reliable data transfer and nearly global coverage. There are
	two cellular standards developed specifically for IoT things.
	LTE-M (Long Term Evolution for Machines) enables devices to
	communicate directly with the cloud and exchange high volumes
	of data. NB-IoT or Narrowband IoT uses low-frequency channels
	to send small data packages.

Once parts of the IoT solution are networked, they still need messaging protocols to share data across devices and with the cloud. The most popular protocols used in the IoT ecosystems are as shown in table 2.3 below:

Item	Explanation
DDS (the Data	which directly connects IoT things to each other and to
Distribution Service)	applications addressing the requirements of real-time systems.
AMQP (the Advanced	aiming at peer-to-peer data exchange between servers.
Message Queuing	SIA MO
Protocol)	
CoAP (the Constrained	a software protocol designed for constrained devices such as
Application Protocol)	end nodes limited in memory and power (for example, wireless
بيا ملاك	sensors). It feels much like HTTP but uses fewer resources
UNIVERS	ITI TEKNIKAL MALAYSIA MELAKA
MQTT (the Message	a lightweight messaging protocol built on top of TCP/IP stack
Queue Telemetry	for centralized data collection from low-powered devices
Transport)	

Table 2.3 Protocol in IoT ecosystem

The processing layer employs IoT platforms to accumulate and manage all data streams and make raw data useful. The processing layer accumulates, stores, and processes data that comes from the previous layer. These tasks are commonly handled via IoT platforms and include two major stages.

The first is the data accumulation stage, where the real-time data is captured via an API and put at rest to meet the requirements of non-real-time applications. The data accumulation component stage works as a transit hub between event-based data generation and query-based data consumption. Next, among other things, the stage defines whether data is relevant to the business requirements and where it should be placed. It saves data to a wide range of storage solutions, from data lakes that hold unstructured data like images and video streams to event stores and telemetry databases. The total goal is to sort out a large amount of diverse data and store it most efficiently.



Figure 2.14 Common architecture of IoT

The second is the data abstraction stage, where data preparation is finalized, as shown in figure 2.14, so that consumer applications can use it to generate insights. The entire process involves the following steps:

- Combining data from different sources, both IoT and non-IoT, including ERM, ERP, and CRM systems.
- 2. Reconciling multiple data formats.
- Aggregating data in one place or making it accessible regardless of location through data virtualization.

Similarly, data collected at the application layer is reformatted here for sending to the physical level so that devices can "understand" it. Together, the data accumulation and abstraction stage veil details of the hardware, enhancing the interoperability of smart devices. What's more, they let software developers focus on solving particular business tasks rather than on delving into the specifications of devices from different vendors.

Lastly, the application layer delivers end-users solutions like analytics, reporting, and device control. The application layer works to address business requirements. At this layer, information is analyzed by software to answer key business questions. Hundreds of IoT applications vary in complexity and function, using different technology stacks and operating systems. Some examples are:

- 1. Device monitoring and control software,
- 2. Mobile apps for simple interactions,
- 3. Business intelligence services, and
- 4. Analytic solutions using machine learning.

Currently, IoT platforms provide software development infrastructure with ready-to-use tools for data mining, advanced analytics, and data visualization. Otherwise, IoT applications use APIs to integrate with middleware.

2.2.1.4 Three-layer IoT architecture

The most basic architecture is a three-layer architecture, as shown in figure 2.15 below. It was introduced in the early stages of research in this area. It has three layers: the perception, network, and application layers.



The most basic architecture is a three-layer architecture, as shown above. It was introduced in the early stages of research in this area. It has three layers: the perception, network, and application layers.

The first layer, known as the perception layer, is the physical layer, which has sensors for sensing and gathering information about the environment. It senses some physical parameters or identifies other smart objects in the environment. This is the physical layer of the architecture. This is where the sensors and connected devices come into play as they gather various amounts of data as per the need of the project. These can be the edge devices, sensors, and actuators interacting with their environment.

Next, the network layer is responsible for connecting to other smart things, network

devices, and servers. Its features are also used for transmitting and processing sensor data. The data that's collected by all of these devices needs to be transmitted and processed. That's the network layer's job. It connects these devices to other smart objects, servers, and network devices. It also handles the transmission of all of the data. The equipment involved at this layer including routers and gateways. for example, where users tap a button in the app to turn on a coffee maker. The technology involve here is cloud storage or cloud server

Lastly, the application layer is responsible for delivering application specific services to the user. It defines various applications in which the Internet of Things can be deployed, for example, smart homes, smart cities, and smart health.



2.2.2 Sensor

The current project makes use of a smartphone's accelerometer sensor. The sensor is already integrated into the motherboard of a smartphone and performs a variety of activities, including measuring the device's acceleration and then tracking various motions such as shaking, tilting, swinging, and rotating, and adjusting the alignment of the custom application accordingly. Additionally, it calculates and detects motion; the accelerometer makes use of the XYZ value, as shown in figure 2.16 below.



Figure 2.16 Three dimensional axes of accelerometer sensor

Accelerometers are a type of electrical sensor that detect the acceleration forces acting on an object in order to determine its spatial location and track its movement. Additionally, acceleration is a vector quantity that is defined as the rate at which the velocity of an item varies. In essence, velocity is the rate of movement of an item divided by the rate of change in time. Acceleration forces are classified into two categories of static and dynamic forces. Static force is constant forces acting on an item, for instance, gravity or friction. Meanwhile, dynamic forces are delivered at varying rates to an object, such as vibration. For example, accelerometers are used in braking systems in automobiles. When a vehicle is subjected to a strong dynamic force, the accelerometer detects a quick deceleration and provides an electronic signal to an input device, deploying the airbags.

Accelerometers are divided into three distinct types, each of which is constructed to perform optimally on a predetermined setting. Piezoelectric, piezoresistance, and capacitive are the three types. First of all, a piezoelectric accelerometer measure changes in acceleration by utilising the piezoelectric effect, which occurs when piezoelectric materials undergo physical stress. Piezoelectric accelerometers are most frequently employed to assess shock and Vibration. Following that, Accelerometers with a piezoresistance sensor are less sensitive than those with a piezoelectric sensor, making them more suitable for transient load testing. The resistance of a piezoresistance accelerometer grows proportionately to the amount of pressure applied to it. Lastly, the capacitive accelerometer is the most often used category of accelerometers. Accelerometers using capacitive sensors determine an object's acceleration by monitoring the change in electrical capacitance. Whenever the sensor is moved, the space between the plates of a capacitor varies proportionally to the diaphragm movement.



Figure 2.17 Diagram of MEMS Capacitive Accelerometer

The majority of accelerometers in a smartphone are tiny; hence it is referred to as Micro Electromechanical Systems or known as MEMS accelerometers, as shown in figure 2.17 above. They are integrated into a wide variety of handheld electronic gadgets due to their tiny size and low costs, such as tablets, smartphones, and video game controllers. The accelerometer in phones and tablets is responsible for "flipping" the screen when the device is rotated. Engineers often use accelerometers to record collision experiments data, and manufacturers monitor machinery vibration.

Accelerometers in motion input act as an individual digital assistant, while accelerometers in digital audio players are used to control the user interface of a mobile application. It alters the mobile application's content and displays orientation to make it more user-pleasant. Additionally, because a smartphone's accelerometer can detect movement, it may be used as a pedometer to tally steps, providing users with detailed analysis of the number of calories burned, the number of kilometers walked, and more in a wearable device. As a result, the sensor is widely used in health and fitness applications as well as sports applications. In terms of motion sensing, the majority of smartphones these days use an accelerometer to adjust the screen orientation based on the direction in which the device is held. Users can improve their viewing experience by using the built-in accelerometer on their mobile phone when playing the game with gesture-based, page-turning, switching from portrait to landscape orientation, and zooming out and in on photos. In general, an accelerometer in a smartphone includes at least one tilt sensor for picture management. Occasionally, it is used to compensate for camera wobble during photography, auto-rotate images, and play motion-sensitive mini-games.

Additionally, accelerometers are used in a wide variety of disciplines, both academic and consumer-oriented. For instance, accelerometers in laptops guard against damage on the hard drive. If the laptop falls unexpectedly while in operation, the accelerometer will detect the unexpected free fall and immediately turn off the hard drive to avoid the reading heads colliding with the hard drive's platter. Without this, that two components would collide, scratching the platter and resulting in serious file and reading damage. Accelerometers are also utilised in automobiles as the industry-standard means of detecting car crashes and instantly activating airbags. Another example is using a dynamic accelerometer to calculate the angle at which a gadget is inclined relative to the Earth. Users can assess how the device is moving by sensing the amount of acceleration.

Accelerometers enable the user to understand an item's environment better. This little device allows it to decide whether an object is traveling upward, whether it will fall over if it is tilted any farther, or whether it is flying horizontally or angled downward. For instance, smartphones' displays rotate between portrait and landscape modes in response to how the phone is tilted.



Figure 2.18 LIS302DL accelerometer sensor

The LIS302DL is a low-voltage three-axis linear MEMS accelerometer, as shown in figure 2.18. It comes with a digital output packed in the form of an LGA packaging. Following that, the LIS302DL is a low-power, super-compact three-axis linear accelerometer that integrates a sensing element with an integrated circuit interface capable of reading data from the sensing element and communicating it to external applications via an I2C or SPI serial interfaces. The acceleration detecting element is manufactured using an ST-developed proprietary process for fabricating inertial sensors and actuators in silicon. Rather than that, using the CMOS process, the IC interface is made, which enables a high degree of fusion to produce a specialised circuit that has been factory-trimmed to match the sensing element's attributes better.

The LIS302DL features a user-selectable full scale of roughly plus and minus of two and eight gram and accelerations that can measure at a 100 Heartz up to 400 Heartz data output rate. Self-testing helps the user to verify that the system is functioning properly. Two independently programmed, highly programmable interrupt sources can be used to generate an inertial wake-up interrupt signal when a programmable acceleration threshold is exceeded along one of the three axes to detect free fall or to recognise single or double click occurrences.

Two separate pins can be configured to provide interrupt signals to linked devices. The LIS302DL is available in a plastic SMD package and is meant to operate over a temperature range that runs from as high as 85°C to as low as negative 40 °C. The SMD package's ultra-compact size and weight make it an ideal solution for handheld portable applications such as mobile phones and PDAs, as well as any other application that requires a small package size and weight.



2.2.3 MQTT Protocol

The current project uses the smartphone application to publish data obtained to the MQTT server. Smartphones use mobile data and also Wi-Fi networks to publish data to the public MQTT cloud broker. Wi-Fi is a relatively new technique of wireless broadband communication that sends multiplexed data over a wide frequency range. Wireless broadband networks are introduced by comparing the geographical population density to the bandwidth limit. Wireless networks are intended to save time and eliminate the numerous obstacles produced by cables, making them more convenient than conventional networking. IEEE 802.11 standards launched Wireless Fidelity, commonly referred to as Wi-Fi technology, in 1997, enabling customers to connect to the Internet from virtually anywhere. However, the infrastructure was costly until 2002, although the latest 802.11g specification in 2003 led to the development of Wi-Fi-activated gadgets for today's mass-market (Surendra Tambe, 2015).

Since its beginnings, Wi-Fi infrastructure has advanced significantly in terms of providing speedier broadband connectivity to Internet applications and data via the radio network, far faster than traditional modems. Radio frequencies such as 2.4GHz and 5GHz rely on wireless hardware such as the Ethernet protocol and CSMA for Wi-Fi technology. As with any networking network, This approach frequently necessitates the use of a transmitter, such as a Wireless Router or Hotspot, and a receiver, which can be any computer equipped with Wi-Fi, such as a desktop, smartphone, and tablet, as illustrated in Figure 2.19 (Surendra Tambe, 2015).



Figure 2.19 Simple Wi-Fi Configuration

Meanwhile, Mobile data is what enables a phone to connect to the internet while the phone is not connected to Wi-Fi. Mobile-enabled gadgets can transmit and receive data via a wireless cellular connection. To be more specific Mobile data is Internet information supplied via a wireless cellular connection to mobile devices such as smartphones and tablets. Cellular service providers have provided mobile data using a variety of various technologies, including GSM in 1G, 2G, 3G UMTS, and 4G LTE Advanced, as well as CDMA and TDMA.

The abbreviation MQTT stands for Message Queuing Telemetry Transport. It is a publish-subscribe messaging system based on the TCP/IP protocol suite. In the year 1999, Andy Stanford-Clark of IBM and Arlen Nipper of Cirrus Link created the protocol's original version. MQTT is faster than HTTP inquiries from the IoT device because MQTT messages can be as small as two bytes in length, but HTTP headers carry a great deal of information that other devices may ignore. Additionally, if there is a significant number of HTTPenabled devices waiting for a request, each client must receive a POST action. When a server receives data from a client, the server transmits the data to all interested clients automatically, as shown in figure 2.20.



Additionally, the data protocol employed in the current project is MQTT-based. MQTT is a protocol designed primarily for "machine-to-machine" communication. The MQTT protocol communicates through TCP/IP and has a small overhead (> 2 bytes), which reduces power consumption. This protocol is data-agnostic, capable of transferring data in various formats, including binary data, text, XML, or JSON. It employs a publish/subscribe architecture rather than a client-server paradigm.

Numerous implementations of the MQTT protocol are available for machines. The well-known MQTT frame must be installed on the device via two major components of software known as MQTT Client. Through the Eclipse and broker of MQTT and Client Library of PAHO, the web portal manages publishing and subscription information using Javascript. A Linux system can make use of a free broker such as mosquito, HiveMQ, or others. (Atmoko et al., 2017).

MQTT is an existing topic that acts as a broker filter, sending messages to all connected and subscribing customers. MQTT implements a quality of service (QoS) standard for functioning output. This degree ensures the communication's reliability throughout transmission. Level 0 messages are transmitted only once. Messages confirming the network's existence are sent with no attempt made to return the message. At a minimum, Level 1 messages are sent once to ensure that the broker receives the client's message status if the subscriber disapproves the request. Level 2 is responsible for ensuring the message is received. At this point, it should be assured that the message is sent successfully and that duplication of received messages is avoided. (Atmoko et al., 2017).

Before begin learn how to design the MQTT network, it is a good idea to familiarise with some of the vocabulary and how each component comes together to form the network as shown below in table 2.4. This fundamental will assist in comprehending and providing a brief overview of MQTT and the concepts associated with it.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

تيكنيكل مليسيا ملا

Keyword	Functions/explanation		
Broker	The broker is the server that disseminates the information to the		
	server's interested customers		
Topic	The message's subject. Clients can contribute to a topic by publishing		
	subscribing or doing both		
	subscribing, of doing both.		
Client	The device communicates with the broker to send and receive data.		
	MALAYSIA 4		
Subscribe	Clients indicate which subject(s) they are interested in. When a client		
2	subscribes to a topic, any message that the broker publishes is delivered		
-	subscribes to a topic, any message that the broker publishes is derivered		
	among the topic's subscribers. Additionally, clients can unsubscribe to		
	stop getting notifications from the broker regarding that subject.		
-	اونيذم سية تتكنيكا مليسيا ملا		
Publish 📋	Clients who submit data to the broker for distribution to interested		
	aliants based on the tonio's name		
	chems based on the topic's name.		
QoS	Service Quality. Each connection can specify the broker's quality of		
	service using an integer value between 0 and 2. The QoS setting has no		
	effect on how TCP data transactions are handled, just on how they are		
	handled between MOTT clients		
	handled between MQTT chefts.		

Table 2.4 Terms used in MQTT

In terms of quality of service, the 0 specifies only once, or only once, and only once without requiring delivery acknowledgement. This tendency is frequently referred to as "fire and forget." At the very least once, as 1 specifies. The message is repeated until it receives an acknowledgement, a procedure known as acknowledged delivery. Finally, for 2, it is specified exactly once. The sender and receiver clients exchange a two-level handshake in order to ensure that only one copy of the message is received, a technique known as assured delivery.



2.3 IoT-Based Condition Monitoring and Fault Detection.

2.3.1 Recent works in IoT-Based Condition Monitoring.

With the accelerated development of the computing power and the eventual deployment of 5G networking networks (S. Li et al., 2018), Combined IoT Software, Cloud computing, and big data modelling systems to forecast mechanical device maintenance are at the forefront of the next growth stage. Wind turbines, gearboxes, rolling bearing, diesel engines, and manual repairs and troubleshooting are usually needed for other mechanical machinery. Therefore, owing to the inability to detect the mechanical device breakdown in time, subsequent health injuries frequently occur (Teng et al., 2019) (Pan et al., 2019). Within an IoT system, it is possible to obtain a significant number of mechanical machines operating data in a short period. Whether to make good use of the gathered data and boost the precision of fault prediction is a difficult problem. Numerous sensors, such as those related to velocity, motion, temperature, and air resistance, have been used in recent years, or various sensors of the same kind have been integrated to gather data on the operating state of different pieces of mechanical machinery in real-time. IoT-based and integrated with cloud framework (Karatza & Stavrinides, 2019) (He et al., 2013), and using multi-source data fusion sensing technologies for large data processing will significantly increase prediction accuracy. This has become a highly important research subject.



Figure 2.21 Topology Diagram of IoT-based Condition Monitoring

The vibration sensor remains an integral feature of the Internet of Things (IoT), driven by the increasingly emerging technologies that increase measuring precision and reduces hardware costs, as illustrated in Figure 2.10. Vibration sensors physically add crucial insight into the operating state of these instruments to central equipment in command and production systems, e.g., motors and tubing.

Firstly, Vibration represents precisely the operating status of the machinery, created mainly with the mechanical action of machinery, for instance as the movement of the motors and the movement of the water in the pipes. Second, Vibration is essentially unaffected by certain external forces, such as temperature along with moisture. Vibration measurements over objective devices provide more accurate info than could be provided by traditional sensors. Third, vibration activity with respect to the ageing equipment is assumed to change rapidly. It is also a perfect predictor of early indicators of faults and issues long before the real problem occurs with the units. Vibration sensors remain developing an important part of IoT in CPS with all these enticing features, particularly to help predictive Maintenance over replaceable components (Jung et al., 2017).

Within currently available Manufacturing Execution Systems or short know as MES. The vibration sensor data in predictive Maintenance involves advanced techniques of collection and analysis. The challenges are double in scale. Firstly, the evaluations obtained via accelerometers extend around a space with a large number of dimensions due to the continuum existence of the vibration data, just part of which is valuable for the objective of maintenance prediction. This high dimensionality requires effective methods of selecting features based on behavioural Vibration spectral properties to identify main metrics to predict precise device failures. Third, the distance between the vibration sensor and the base is limited due to the sensors' hardware and strength constraints. Following that, an analytical rear-end motor is necessary. To carry out sporadic and unfinished measurements since several sensors were used for quantitative modelling.



Figure 2.22 Usage of a vibration sensor

On-site diagnostics of the safety state of the pump can be performed directly by examining vibration data after the spinning motor within the pump in real-time. However, no possibility for disassembling the pump in order to mount up the sensor that detects Vibration during its operation. The proposed IoT-driven method is instead to track continuously the vibrations propagated on or after the engine in relation to the external force connector through a non-intrusive way (Jung et al., 2017).


2.3.2 Vibration-Based Fault Detection

Machines consist of moving components that generate sound and Vibration. Each component will have a different vibration signal depending on the condition and construction of the unit. The vibration signal varies as well as the machine's part state changes. This adjustment to the signature of the Vibration shows the original defect and can thus be observed and repaired before failure. This is the biggest advantage of situation control. The control of the condition can improve maintenance performance and the likelihood of major injuries and can also save money. This vibration analysis is a crucial instrument to track the use of electrical devices, transducers, processors, and applications fully automatic system monitoring. This paper provides a study of the varied diagnostic strategies for revolving mechanical parts.

The vibration technique has two primary goals. The initial step is to separate the system signal from other components and reduce noise early. Secondly, the state of the system and the faulty parts are determined. Spectral analysis, order analysis, time-synchronous average, time-frequency analysis, waveform analysis, Fast-Fourier Transform (FFT), and probability density instances are all examples of this type of analysis. Which have part of signal processing methods, all come with several techniques. This vibration-based diagnostic method is commonly used due to its ease of calculation. Vibration analysis has formally been used to assess the failure and critical operational conditions (Kumar et al., 2018).

• Techniques of Condition Monitoring

The technique of conditional monitoring is applied to detect major modifications to the detected equipment. These improvements suggest the fault's growth. This is a method of statistical repair.

• Detection of fault and Diagnosis via Vibration

In operating conditions, the signal of Vibration produced by the computer includes the consequence for numerous distinct components and the noise. The signal relevant to the component under observation must be identified and selected. The vibration signals are usually obtained from the housing. At the beginning of the production process, the presence and form of fault are detected, its progress is tracked, and the rest of the system life is presumed. This helps plan the proper repairs. Typical causes of vibrations in belt drives are vertical, lateral, and loose belt situations (Jagtap, 2017).

• Techniques of Signal Processing

Signal processing is required for the condition control system. Also, various kinds of defects; thus, it is important to process these signals. The approach chosen is determined by the signal's quality.

Based on the literature study discussed above, most machine condition monitoring uses Vibration as a physical parameter to be measured. The measurement process is proposed by using IoT-based sensors and devices. Thus, the current project aims to develop an IoT device capable of measuring a rotating machine's acceleration signal and developing a warning system to notify the user or operator should the machine enter a faulty condition (Abdulshakoor et al., 2015).

Vibration monitoring is an efficient and rapid method for diagnostic failure. Vibration signals are often obtained at a low level of sound that is either from machines or the environment. Using vibration analysis, mechanical and electrical faults such as stator faults, rotor, misalignment, transmission machine faults, and bearings are identified (Delgado-Arredondo et al., 2017).



2.4 Summary

This chapter presents IoT technologies on this project and discusses sensors in smartphones and their uses. Also, regarding MQTT workflow and how the project use or implements IoT technologies in monitoring and faulty detection. Besides that, many IoT-based state monitoring uses Vibration as a physical factor to determine the health status of the rotating machine.



CHAPTER 3

METHODOLOGY

3.1 Introduction

Chapter 3 discuss the methodology used in the current project. There are three significant steps in the present work: MIT application designs, data collection, and data validation. This chapter also discusses the process of implementing the predictive maintenance concept to the rotating machine. A rotating machine in UTeM's laboratory was used to collect the vibration data. The data collection and information processing are conducted using the developed IoT application using a smartphone.

This chapter's topic is mainly concerned with data collection. For instance, the vibration of a rotating machine is monitored using an accelerometer sensor, and the data is uploaded to the cloud, where it is translated to a signal and used to detect a problematic state in real-time using coding and machine learning technologies. Additionally, data validation is carried out by comparing the frequency of the rotating machine to the frequency measured by the IoT device. This chapter also discusses identifying machine faults and failures using predictive maintenance concepts and IoT-based warning systems.

3.2 Research Design and Method



Figure 3.1 Research Methodology

Figure 3.1 above indicates the research approach used in the current project. Essentially, this research is divided into three major sections or layers. The first is the Internet of Things philosophy, which is primarily concerned with data collection. The device used is a smartphone with an accelerometer sensor and a developed MIT application. This is because the accelerometer sensor is suitable for reading the movement on all axes and transferring the data to the IoT platform. Besides that, in this project, the data obtained will be sent over to the MQTT broker cloud in real-time. The IoT device will be attached to the machine to detect the health status of the machine, and it will detect abnormality in the operation of the machine.

This project uses an accelerometer sensor built-in smartphone and using developed applications. How it works is that it will read the data produced and sent to the cloud server in real-time. The application is developed and programmed by MIT application inventors. MIT Application inventors are platforms that are based on website software programs. The integrated development environment formerly accommodated by Google is now maintained by the Massachusetts Institute of Technology, often known as MIT. Through the use of a web browser, the connected phone, or an emulator, Application Inventor enables the user to design applications for Android phones. Application Inventor servers save work progress and aid user with project management.

Additionally, at MIT, users can create an application by utilising The Application Inventor Designer, which allows users to select their application's components. The Application Inventor Blocks Editor enables the user to create programme blocks that specify the behaviour of the components. User arranges the program blocks of code visually as if they were puzzle pieces. User application will appear on the phone in stages as User add elements, allowing the user to test and troubleshoot their work as they go. After finished, package the program and create a self-contained program for installation on the device. Even without owning an android phone, it is still possible to develop applications via an Android emulator, which works on a computer with functions identically to an Android phone. App Inventor runs on GNU or Linux, Windows, Mac OS, and a variety of notable Android smartphone variants. Next, applications developed with the Application Inventor platform are able to be installed and run on any Android smartphone.

Moving on, the accelerometer is used to detect Vibration on the machine, which is Vibration using axis X, Y, and Z. The axis value can identify if the machine is in a regular operation or faulty operation. Furthermore, after collecting the data, the application will upload the data to the public MQTT cloud broker to further analyse and visualise the data. Besides that, the data will be analysed in MATLAB by using signal processing techniques called Fast Fourier Transform or FFT in short. The FFT translates data in the time domains to input in the frequency domains. A Fast Fourier Transform or short-form as FTT is a mathematical operation that computes the discrete Fourier transform (DFT) or its inverse of a sequence (IDFT). The Fourier transform is used to convert a signal's native domain, which is typically time or space, to a frequency domain representation and vice versa

3.3 Design of Experiment

3.3.1 Application Building and Development



Figure 3.2 Work Flow of SENSE IT application developed

From Figure 3.2, we can see the flow of Applications that have been developed using MIT app inventor. First, the application will read acceleration data from a built-in accelerometer in an Android-based smartphone. After that, the application will publish the data to the MQTT cloud broker such as HiveMQ MQTT cloud broker. Finally, node-RED will subscribe the data from the

MQTT cloud broker, and there it will be displayed in graphical form and able to be saved as comma-separated values (CSV) file.



The full flow of the SENSE IT application is shown in Figure 3.3, where it starts with the machine and ends with actions. Firstly, the IoT device is placed on the machine. The IoT device is a smartphone with an accelerometer sensor built in to collect the data and send it to the MQTT cloud broker. Then the data will be subscribed by Node-RED, where data could be shown in a dashboard graphically, and there is also a warning system in the node-RED flow, which will be triggered when abnormal conditions occur to the machine. When the abnormal condition occurred the email will be automatically sent, and then action will be taken according to the situation.

3.3.2 Data Collection



Figure 3.4 Data Collection Framework

Figure 3.4 shows the framework for data collection. The first step is to mount the smartphone or known as IoT Device, at the machine. Careful consideration must be given during the attachment of the IoT device in the machine so that the desired frequency can be adequately captured. The selected machine for data collection is a rotating machine that has a cyclic vibration characteristic. The second step is the Application setup and publishing data to the public cloud MQTT broker. After that, the data is subscribed and connected from the MIT application to the cloud of the MQTT broker.

Before the data subscription, the flow in the MIT application needs to be prepared in the first place. They are using an extension in the MIT application that is UrsAI2MQTT to connect the application to the cloud server so that the data can be retrieved from the public cloud MQTT broker. This will connect the application and the data produced with the accelerometer to the cloud broker server, sending the data in real-time. Besides that, the data collected by the accelerometer sensor will be on three-axis which are X-axis, Y-axis, and Z-axis. The data will be subscribed by Node-RED from the MQTT cloud server there; it will be shown graphically and able to be saved. Next, the data need to be addressed to the correct location on the computer. The data will be saved in CSV format (Comma-Separated Values); it can be open using an application, for example, Microsoft excel.



Figure 3.5 shows the flowchart for data collection. The data collection begins by turning on the rotating machine. Then the accelerometer sensor in the smartphone measured the acceleration of the machine, and the data appeared in the applications. After that, the obtained data is then sent to the cloud every 0.05 sec. The MIT application software that sends the data is subscribed to the cloud server, and then the data received is subscribed by node-RED and finally stored in the form of CSV format. In the MIT application, the extension must be added UrsAI2MQTT extension, which will subscribe to the generated data on smartphones on the MQTT server. There is also the option either to turn on or off the notification on the MIT application.

3.3.3 Data Validation



Figure 3.6 Data Validation Framework

The data validation approach used in the current research is depicted in Figure 3.6. Firstly, the machine is set to the desired frequency of 10hz,15hz or 20hz, the data collection of the time domain is started after the application is set on the machine, and another setting is done. The data from the time domain will be saved in CSV(comma-separated values) format; this data is then refined to 1 minute and loaded up into MATLAB application in order to run the transformation of the data obtained using the Fast Fourier Transform method to acquire a representation of data in the form of the frequency domain. The Fourier Transform (FFT) is a mathematical technique for transforming a continuous signal between time and frequency domains. The final data frequency is then compared with the machine's initial set frequency to check and validate the data obtained is the same and valid, as shown in figure 3.7 below.



Figure 3.7 Process of data validation

3.3.4 Real-time Warning System



Figure 3.8 Real-time alert system

Figure 3.8 shows the real-time alert system for abnormalities that occur from the rotating machine. Firstly, data is sent from the SENSE IT application to the MQTT cloud server then the

data is subscribed into Node-RED, where there is a node to monitor if any abnormalities occur. If abnormalities occur, the alert system is triggered, and an email is sent directly in a real-time manner to warn about the abnormalities that occur. The simple warning system flow is shown below in figure 3.9.



Figure 3.9 Work flow for warning system alert

3.4 Summary

This chapter is about the methodology of this project, such as data collection, data analysis, data validation, and design of IoT MIT application. The data collection starts with the connection between the MIT application and cloud MQTT. Next, using MATLAB to visualise the data. The data validation is frequencies from the IoT unit can be measured and then compared to the machine's frequency and have the same frequency value. Lastly, step in creating an IoT device application by MIT application platform, IoT AI extension website, and MQTT broker server.



CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

Chapter 4 discusses the application building and development, which shows the application sense it is developed and explains in detail each part, including design and architecture. After that, MQTT data published and subscribe is explained in detail regarding how it works. Moving on, the data collection process which shares about measurement procedure and measurement of data process. In addition to that, Model training of 3 cases is also discussed alongside performance analysis, including test results and Fast Fourier Transform result.

ونيوم سيتي تيكنيكل مليسيا ملاك UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.2 Application Building and Development

4.2.1 Application Design and Architecture

4.2.1.1 Introduction



Figure 4.1 Logo of the SENSE IT Application

First and foremost, as shown in figure 4.1, the SENSE IT application was developed as a final year project titled IoT-based condition monitoring of rotating machinery and predictive maintenance. The intention is to fulfill the research objective to design and develop an IoT (Internet of Things) platform for a machine monitoring system. The application is developed and programmed by using MIT application inventors, as shown in figure 4.2. MIT Application inventors are platforms that are based on website software programs and use a code-block, as shown below in figure 4.3. The integrated development environment formerly accommodated by Google is now maintained by the Massachusetts Institute of Technology, often known as MIT. Through using a web browser, the connected phone, or an emulator, Application Inventor enables the user to design applications for Android phones.

\leftarrow \rightarrow C \textcircled{a}	O A https:/	//appinventor. mit.edu				습	S IN ID 💕	Ⅲ ■	× *	=	
- 🔮		Apps! About	Educators	News Reso	aurces Blogs	G	Sive ENHANCED BY G	00	2	^	
With MIT App Inventor, anyone can build apps with global impact											
_	Active Users today: 114.5K	Active Users this week: 355.5K	Active Users this month: 917.7K	s Register Users 8.2M	red Cou : 1	ntries: 195	Apps Built: 34.0M				
	Contractions	Join the MIT Ap Click here to lea	p Inventor Appath	non for Good 2021			,			v	

Figure 4.2 Home page of MIT application inventor



Figure 4.3 Code-blocks on the application inventor

4.2.1.2 The application interface



Figure 4.4 Logo of SENSE IT on smartphones

First, the application has been designated with a specific logo and name, as shown above. Name SENSE IT was inspired as the application works with a sensor to sense a problem or abnormal condition in rotating machinery. The logo of SENSE IT resembles a vibration and noise from a vibration signal, as shown in figure 4.4. The application's home screen consists of 7 clickable buttons with specific tasks and functions, as shown in figure 4.5 below.



Figure 4.5 Home screen of SENSE IT Application

Table 4.1 below shows each of the button names and the detailed function of the button and explanation.

Button name	Function					
Open Menu	The function is to open a side menu consisting of					
	information about the Internet of Things (IoT),					
	Artificial Intelligence(AI) and predictive					
	maintenance(PdM)					
Close Menu	To close the side menu screen and return to the home					
	screen					
Accelerometer	The page that consists of the setting for accelerometer					
	such as turning on and off, connecting to MQTT					
SAL MALAYSIA MEL	broker and subscribing the value of the accelerometer					
MQTT connection	The page that's connected to various types of MQTT					
L LING	broker					
Graph	The real-time moving graph of acceleration for X-					
ىل مليسىيا ملاك	axis, Y-axis and Z-axis					
About UNIVERSITI TEK	A brief explanation about the application, such as why					
	it is developed					
Exit	The button to exit and close the application					

Table 4.1 Button name and function on the home screen of the Application

4.2.1.3 Open menu button

When this button is pressed, the side menu screen will pop up with a few selection options, as shown below in figure 4.6.



This menu is created to educate and enrich the knowledge of the application user. The side menu consists of information about the Internet of Things (IoT), Artificial Intelligence(AI) and predictive maintenance(PdM). This info consists of basic definition, the use-case, advantage and even disadvantages as shown in figure 4.7.



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.2.1.4 Close menu button

This button function is very simple to close the info page and back to the home screen, as illustrated in figure 4.8 below.



4.2.1.5 Accelerometer button

The accelerometer section is the most important part of the application. The main highlight is which sensor or accelerometer reading is taken and sent over to the MQTT broker in real-time. The interface screen is below in figure 4.9.



As seen above, there is 2 section which is the Acceleration sensor, which consists of an on/off button used to start or stop the accelerometer value. Also, the acceleration value reading of X-axis, Y-axis and Z-axis. Besides that, it also consists of a reset button that will clear the accelerometer's reading. Meanwhile, the MQTT setup section consists of connecting, disconnect and subscribe values, and lastly, the back button that functions to return to the application's home screen. The detailed Accelerometer sensor and MQTT setup as shown in table 4.2 below.



Table 4.2 Accelerometer sensor and MQTT setup









4.2.1.6 MQTT Connection button

This section is created so that the user can connect to the different cloud brokers with their info to obtain the data with no issue safely. There are many cloud brokers such as Mosquitto, Mosca, EMQ or even local IoT platforms favorIoT. With this section, users can send real-time acceleration data to any cloud MQTT platform that is available. The interface is as shown below in figure 4.10



Figure 4.10 Multiple platform connection option

Table 4.3 below shows the public MQTT setup consisting of MQTT URL textbox, Client ID textbox, Username Textbox, Password Textbox, Connect to MQTT broker button and others.

Table 4.3 Public MQTT setup





	×11401 😤 26	66 B/s	1:56 AM	🕰 🕒 😇 84% 🔳			
		PUBL	IC MQTT SETUP				
		MQTT URL :	broker.hivemq.cor	n			
		CLIENT ID :	clientid-xOepTtYu	9			
		USERNAME :	naz2021				
	,	PASSWORD :					
		CONNEC	T TO MOTT BROKE	2			
		START	ACCELERATION- JBLISH-SEND				
		PASS	WORD IS SET				
Connect to MQTT broker	The button used to c	onnect	to the MQ	TT brok	ker that the user		
button	has chosen using the data provided						
ALAYS	1. After the information required is filled, the connect button						
Of the	is clicked, and the image will show up, which shows the						
Kent I	is chece, and the image will show up, which shows the						
	connection is success	sful.		VI			
The a	<u>*11 역</u> 기 중 32 KE	3/s	2:18 AM	A 🕑 😇 82%	5 ()		
"AININ		PUBL	C MQTT SETUP				
et la	ulo 1 Sei Ma	TT URL :	broker.hivemq.c	om			
	CLI	ENT ID.	clientid-xOepTt	′u9			
UNIVERSI	TI TEKNIK <mark>AL M</mark>	ERNAME :	naz2021	LA <mark>K/</mark>	1		
	PAS	SSWORD :	•••••				
	C	ONNECT TO	MOTT BROKER		_		
		START ACC	ELERATION-PUB	ISH-			
Start acceleration-publish-	The button consists of	of 3 fun	ctions is to	start th	e accelerometer		
send button	reading of the X-ax	xis, Y-	axis and	Z-axis.	Also, it is for		
	publishing and sendi	ng the a	lata from a	cceleror	meter reading to		
					in the reading to		
	the MQTT cloud.						


Subscribe-receive button	After clicking, the subscribe value will display the X-axis, Y-
	axis and Z-axis, and a pop-up will appear that says successfully
	subscribed the value from the MQTT cloud.
	1. After the button clicks, the value is sent to the cloud broker
	and subscribed to the application again. When stoped, this
	value will be equal, which shows the last value sent is the last
	value received
	SUBSCRIBE-RECIEVE RESET
MALAYS	ACCELEROMETER VALUE SUBSCRIBE VALUE
TERING	0.67 0.67 -0.203 9.787 9.787 BACK
Reset button	The reset button is used to clear the value and reading of the
dina :	accelerometer value and subscribe value
سيا ملاك	اويومرسيتي تيكنيكل ملي
UNIVERSI	1. when the reset button is clicked, the value of acceleration
	and subscribe value of X-axis, Y-axis and Z-axis are cleared
	RESET
	ACCELEROMETER VALUE SUBSCRIBE VALUE BACK
Accelerometer value	Accelerometer value is the value of real-time acceleration that
	reads from the accelerometer sensor.





4.2.1.7 Graph Button

The graph button section is where a real-time graph is generated from the accelerometer's reading on the device. The graph will show the acceleration reading in the range of 10 to -10. The interface is quite simple with only three buttons: start, stop, and back button, as shown in Figure 4.11 below.



Figure 4.11 Interface of the graph section

The button function on a graph section is explained in detail in table 4.4 below for the start and end buttons.



 Table 4.4 Button function on a graph section

Васк	The button will send the user back to the home screen of the application

The graph is developed using HTML code to show the data collected from the accelerometer plotted into a real-time graph of x-axis, y-axis and z-axis, as shown in figure 4.12.

کل ملیسیا ملاط	يكنيه	المن مسيني تر Add to Fav	New Save & Share
<pre>Sample ③ D D D D D D D D (Constraint)</pre>	File	Output Accelerometer ELAKA 10 0 0 0 0 0 5 10 5 10 5 10	^
<pre>15 - (body) 16 17</pre>			×

HTML Viewer Online

Related Tools

Figure 4.12 HTML code used to generate-graph

The HTML code has been written to produce the output of plotting a real-time acceleration graph. This HTML code is connected to the google live graph extension; thus, it updates the accelerometer data of obtained value in real-time.



4.2.1.8 About button

The about section is created to give info regarding the application, such as why the application is developed and the reason and goals of developing it. The interface is shown in Figure 4.13 below.



Figure 4.13 Interface of the about section, which explains the SENSE IT application

4.2.1.9 Exit button

Figure 4.14 below shows the exit button, and its function is very simple: to exit and close the application.



Figure 4.14 Exit button image



4.2.2 MQTT Data Publish and Subscribe

4.2.2.1 HiveMQ MQTT Platform Background Introduction

HiveMQ was founded in 2012 and is based in Landshut, Germany, near Munich. HiveMQ, the company's flagship product, is a MQTT-based messaging platform optimised for the quick, efficient, and reliable bi-directional transit of data between devices and the cloud. With products such as HiveMQ Cloud and HiveMQ Swarm, the company's portfolio is continually expanding.

The platform for the MQTT cloud server is HiveMQ MQTT cloud broker, as shown in figure 4.15 below. HiveMQ is a MQTT broker and client-based messaging platform that enables the rapid, efficient, and reliable transfer of data between linked IoT devices. It makes use of the MQTT protocol to provide bi-directional data push between devices and corporate systems in real-time.



Figure 4.15 Example of devices and enterprises that connect to the HiveMQ MQTT platform

HiveMQ is designed to address several of the most significant technical challenges organisations face when developing new Internet of Things applications, including developing reliable and scalable business-critical IoT applications, delivering data quickly enough to meet end-user expectations for responsive IoT products, lowering operational costs through efficient use of hardware, network, and cloud resources, and integrating IoT data into existing enterprise systems.

HiveMQ enables businesses to connect their equipment to the Internet. HiveMQ enables the secure, dependable, and scalable transfer of data from device to cloud. HiveMQ envisions a connected society where individuals and businesses can achieve their full potential. Over 130 customers, including Fortune 500 businesses, rely on HiveMQ technologies for mission-critical use cases like linked automobiles, logistics, Industry 4.0, and connected IoT items, as illustrated in Figure 4.16.



Figure 4.16 Several key organisations that joined the HiveMQ community

4.2.2.2 Reason and Advantages of Using HiveMQ

There are a number of reasons to use the HiveMQ platform. It is an open-source platform that was unveiled in 2011, and they have specific open-source goals, such as increasing the usage of high-quality MQTT implementations. The HiveMQ Community is expected to be ideal for learning, proofs of concept, research projects, and small-scale installations for individuals. For mission-critical deployments, the majority of customers prefer a commercial, fully supported solution such as HiveMQ Professional or HiveMQ Enterprise. Following that, they seek to simplify the usage of HiveMQ and MQTT by HiveMQ partners, students, academics, and other open-source initiatives. This, we think, will result in increased value for the entire ecosystem, including HiveMQ and our clients. Finally, our commercial software offerings will incorporate open-source functionality and upgrades, allowing our clients to benefit from the contributions of a broader developer community.

Table 4.5 below shows the benefits of using the chosen HIVEMQ MQTT platform alongside with detailed explanation.

Table 4.5 Benefits of the HiveMQ MQTT platform

Reason	Detailed explanation	
Scalable MQTT	Instances of HiveMQ MQTT broker scale in lockstep with the underlying	
Broker	hardware. The non-blocking and multi-threaded technique enables	
	simultaneous device connections of up to ten million while retaining	
	extraordinarily high throughput and introducing negligible latency.	
Reliable Data	It might be difficult to deliver data across unreliable networks. HiveMQ	
Delivery	supports all MQTT Quality of Service levels, including delivery at most	
	once, delivery at least once, and delivery exactly once. To compensate	
	network latency, HiveMQ's support for complex message retention policies	
.5	and offline message queuing is critical.	
Elastic	It might be difficult to deliver data across unreliable networks. HiveMQ	
Clustering	supports all MQTT Quality of Service levels, including delivery at most	
0.00	once, delivery at least once, and delivery exactly once. To compensate	
لاك	network latency, HiveMQ's support for complex message retention policies	
and offline message queuing is critical.		
Enterprise-grade	e HiveMQ is intended to protect IoT data as it travels from device to	
security	enterprise systems. Industry-standard security measures such as TLS 1.3,	
	secure WebSockets, and state-of-the-art cypher suites are used to transfer	
	data. X.509 certificates, username/password, and IP-based authentication	
	are all supported, as is an API that enables bespoke authentication,	
	authorisation, and permission logic, such as OAuth 2.0 integration.	
100% MQTT	HiveMQ's MQTT broker complies with the MQTT 3.1, MQTT 3.1.1, and	
Compliant	MQTT 5 specifications to the letter. Additionally, we support concurrent	

	communication with HiveMQ using MQTT 3 and MQTT 5. At scale, all
	advanced capabilities like as topic wildcards, persistent sessions with
	offline queuing, preserved messages, and service-level agreements are
	available.
Deploy	HiveMQ is available as a private, hybrid, or public cloud service. Pre-built
Everywhere	images can be deployed using Kubernetes, OpenShift, or DC/OS on private
	clouds. AWS and Microsoft Azure are two public cloud systems that are
	supported. Additionally, HiveMQ can run natively on Linux, Windows,
	and OS X
Efficient	Unlike HTTP, HiveMQ and MQTT use a pub-sub architecture, which
Network	reduces overall network bandwidth by eliminating client polling. MQTT
Utilization	messages are also substantially less in size than HTTP, which reduces the
11102	quantity of data flowing via the network.
Enterprise Data	Enterprise data integration is accomplished by bidirectional data exchange
Integration	between a HiveMQ MQTT broker and an enterprise system acting as a
@Scale UNI	MQTT client. Each enterprise system's MQTT client subscribes to the data
	that has to be integrated via the MQTT pub-sub protocol. HiveMQ's MQTT
	Shared Subscription implementation enables horizontal scaling of MQTT
	clients, ensuring scalable and dependable enterprise connectivity.
Real-time Data	Administrators may monitor real-time data going via the MQTT broker and
Monitoring	MQTT clients linked to your IoT application using the HiveMQ dashboard.
	An administrator can obtain a 360-degree view of each MQTT client's
	client status, disconnect a client, terminate the MQTT session, and
	add/remove subscriptions. HiveMQ enables sophisticated troubleshooting

	by enabling the creation of trace recordings that may be used to identify
	issues and bottlenecks in your deployed IoT applications. A summary
	dashboard provides an operation team with a complete real-time view of
	the broker cluster and the overall health of the system.
Extension	Integrating HiveMQ and your IoT data into existing enterprise systems is
Framework and	made feasible by an open API and a flexible extension mechanism. The
Marketplace	extension framework enables developers to rapidly design bespoke data
	processing, device authentication, and device authorisation extensions.
	Additionally, HiveMQ features a marketplace for pre-built extensions for
	Kafka, Oracle Database, MongoDB, and other systems.
MQTT Client	HiveMQ supports the use of any MQTT-compliant client library. HiveMQ
Libraries	ships with its own Java client library, but you can also use libraries like
LIS	Eclipse Paho, C/C++, JavaScript, or Python. Additionally, some of our
	customers develop their own proprietary MQTT clients. Multiple MQTT
لاك	client alternatives ensure that you are not tied to a single provider.
Fully Managed	HiveMQ Cloud is a cloud-native IoT messaging service that simplifies
MQTT Cloud	MQTT platform deployment and management. With a few clicks, our fully-
Service	managed MQTT Cloud Platform delivers scalable and dependable MQTT
	cloud-broker clusters that are ready for production.

4.2.2.3 Using Public Broker of HiveMQ Platform

HiveMQ is a public MQTT broker that anyone can use. Users may create their own MQTT clients to connect to the broker. HiveMQ includes a dashboard that allows users to monitor the broker's traffic volume. Additionally, HiveMQ maintains a list of MQTT client libraries for usage with HiveMQ. HiveMQ is accessible via the broker URL broker.hivemq.com. Meanwhile, TCP Port 1883 is open, whereas WebSocket Port 8000 is closed. After filling in details such as login, password, publish topic, and subscribe topic, the MQTT web client will display the message from the device as shown in figure 4.17.



Figure 4.17 MQTT browser client interface

The username and password are used in order to secure the data, as shown in figure 4.18. This is to ensure only authorise person can obtain the data and make sure the data is not leaked to any third party. This is important as the main problem in deploying IoT is security.

Components	Properties
😑 🛄 Horizontal Arranger	UrsAI2MQTT1
Button1_CONNE	Broker
Mage1	broker.hivemq.com
Button2_DISCONNI	ClientID
Button5_SUBSCRIE	
TableArrangement3	loTimeout
A Label2_SUBSX	5
A Label3_SUBSY	KeepAlive
A Label4_SUBSZ	5
Button1_BACK	Password
AccelerometerSensor ⁻	
UrsAI2MQTT1	Port
A Notifier1_CONNECT	1883
A Notifier1_DISCONNEC	UserName
A Notifier1_subscribe_v	naz2021
Sound1_pop	
Sound2_mouse_click 🗸	
< >	

Figure 4.18 UrsAI2MQTT extension where broker, password and username are setups

Table 4.6 below shows a detailed explanation of how to use and set up the HiveMQ

broker platform

اونيۆم سيتي تيكنيكل مليسيا ملاك

Table 4.6 Step to use HiveMQ broker platform

Step	Explanation	
Visit the website of MQTT at	Truduct (1)	Cived Developers MCTT Boldons Blog Company (intrinsicil) (2nctorin)
https://www.hivemq.com/public- mqtt-broker/	<section-header><image/><image/><text><text></text></text></section-header>	MQTT Browser Client

Fill in the username and	Connection
password and click connect	Host Port ClientID Disconnect B000 clientId-NO105GLUn Disconnect Username Password Keep Alive SSL Clean Session Az2021 60 ClientId-NO105GLUN ClientId-NO105GLUN Last-Will CoS Last-Will CoS Last-Will CoS Last-Will CoS Last-Will Retain 0 ~
After that, subscribe to the topic name	Color QoS 2 Subscribe Topic SENSE Subscriptions
Then write the topic name and	Publish
click publish	Topic QoS Retain SENSE 0 Publish Message
After that, the data will be	Messages ×
published in real-time	2021-10-10 05 03 03 Topic: SENSE Gos: 0 {"X-AXIS": 0.421, "Y-AXIS": 5.937, "Z-AXIS": 7.757} 2021-10-10 05 03 03 Topic: SENSE Gos: 0 {"X-AXIS": 0.476, "Y-AXIS": 6.129, "Z-AXIS": 7.489} Gos: 0 2021-10-10 05 03 03 Topic: SENSE Gos: 0 {"X-AXIS": 0.632, "Y-AXIS": 6.129, "Z-AXIS": 7.489} Gos: 0 2021-10-10 05 03 03 Topic: SENSE Gos: 0 {"X-AXIS": 0.517, "Y-AXIS": 6.11, "Z-AXIS": 7.374} Gos: 0 2021-10-10 05 03 03 Topic: SENSE Gos: 0 {"X-AXIS": 0.517, "Y-AXIS": 6.071, "Z-AXIS": 7.374} Gos: 0 2021-10-10 05 03 03 Topic: SENSE Gos: 0 {"X-AXIS": 0.536, "Y-AXIS": 6.071, "Z-AXIS": 7.374} Gos: 0 2021-10-10 05 03 03 Topic: SENSE Gos: 0 {"X-AXIS": 0.536, "Y-AXIS": 6.071, "Z-AXIS": 7.374} Gos: 0 2021-10-10 05 03 03 Topic: SENSE Gos: 0 {"X-AXIS": 0.536, "Y-AXIS": 6.071, "Z-AXIS": 7.568} Gos: 0 2021-10-10 05 03 03 Topic: SENSE Gos: 0 {"X-AXIS": 0.536, "Y-AXIS": 6.052, "Z-AXIS": 7.508} Gos: 0 2021-10-10 05 03 03 Topic: SENSE Gos: 0 {"X-AXIS": 0.5593, "Y-AXIS

4.3 Data collection

4.3.1 Introduction to Vibration analysis

The critical aspect of vibration analysis is that it enables you to monitor and detect problems through the use of vibration data. Read about the methodology, equipment, and techniques used in vibration analysis and vibration analysis measuring methods.

Vibration analysis is described as the process of determining the vibration levels and frequencies of machinery in order to determine the health of the machines and its components. While the internal mechanisms and algorithms for calculating various types of vibration can become intricate, it all begins with the usage of an accelerometer to sense vibration. When a piece of machinery is in operation, it generates vibrations. An accelerometer attached to the machine generates a voltage signal that indicates the amount and frequency of vibration produced by the machine, which is typically expressed in terms of how many times per second or minute the vibration happens.

All data captured by the accelerometer is sent straight to a data collector (software), which

All data captured by the accelerometer is sent straight to a data conector (software), which captures the signal as amplitude versus time (referred to as time waveform), amplitude vs. frequency (referred to as fast Fourier transform), or both. All of this data is analysed by computer algorithms, which engineers or skilled vibration analysts then analyse to determine the machine's health and identify potential problems such as looseness, imbalance, misalignment, and lubrication concerns. Vibration analysis can be used to detect a variety of issues, including the following as listed in table 4.7:

	No	Method of vibration analysis
	1.	Imbalance
	2.	Bearing failures
	3.	Mechanical looseness
	4.	Misalignment
	5.	Resonance and natural frequencies
	6.	Electrical motor faults
	7.	Bent shafts
	8.	Gearbox failures
And I H	9.	Empty space or bubbles (cavitation) in pumps
TEKA	10.	Critical speeds
THEFT	(wn	
للك	• [1	ونيوم سيتي تيڪنيڪل مليس

Table 4.7 Method of vibration analysis

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.3.2 Vibration Analysis Methodology

While accelerometers are still the most common tool used to collect vibration data, modern technology and improved sensor technology have allowed for non-contact, high-speed laser sensors that can detect issues accelerometers can't. This allows for a more accurate and more localised analysis and opens up vibration analysis to more methodology. Vibration analysis is generally broken down into four principles, and each principle gives you specific information on the working conditions and features of the vibrating parts. While accelerometers remain the most often used instrument for collecting vibration data, advances in sensor technology have enabled the development of non-contact, high-speed laser sensors that can detect faults that accelerometers cannot. This enables a more precise and localised examination and broadens the methodology available for vibration analysis. Vibration analysis is often divided into four concepts, each of which provides detailed information about the operating circumstances and characteristics of the vibrating parts.



Figure 4.19 Example graph of time waveform analysis

Table 4.8 below shows the four basic vibration analysis principles alongside the detailed explanation.

Principle	Explanation
Time-	A waveform is created when a vibration signal is captured by a transducer (the
domain	device that converts a physical quantity to an electrical signal) and displayed
	on an oscilloscope screen. This is a time-domain signal. The time domain is
	defined as the amplitude displayed against the passage of time. While the
	majority of machine vibration concerns may be discovered using spectrum
	analysis, certain types are more easily identified utilising waveform analysis.
Frequency	When the previous waveform is subjected to spectrum analysis, the resulting
domain	image of frequency versus amplitude is referred to as a spectrum. The spectrum
5	is a frequency domain representation, just as the vibration is a time-domain
	representation. The majority of in-depth analysis of equipment vibration occurs
	in the frequency domain or through the use of spectrum analysis.
Joint	Due to the fact that vibration signals fluctuate over time, computing many
domain	spectrums at the same time can be advantageous. Thus, a technique known as
	Gabor-Wigner-Wavelet can be used. This technique is used to compute variants
	of the fast Fourier transform (described further below), such as the short-time
	Fourier transform (STFT)
Modal	The modal analysis process converts a piece of machinery's observed frequency
analysis	response functions to a computer model. The computer model can be shown
	with animations demonstrating all of the various vibration modes. The model
	can be changed by adding or subtracting mass or stiffness to observe the effects.

Table 4.8 Four basic principles of vibration analysis

The power spectral density function (PSD) depicts frequency-dependent changes (energy) strength. In other words, it indicates which frequencies have high variations and which have mild variations, as illustrated in figure 4.20 below.



Figure 4.20 Example graph of frequency-domain vibration signal

Apart from these four fundamental principles, which provide precise information on vibrating parts' operating circumstances and characteristics, countless other types of analysis, calculations, and algorithms are utilised to identify various elements of vibration analysis. These include the following as listed in table 4.9 :

Table 4.9 List of forms of analysis, calculations and algorithms used to determine different aspects of vibration analysis

Forms	Explanation	
Time waveform	A time waveform is a representation of acceleration versus time in the form	
	of tables and charts. Time waveforms depict a brief snapshot of raw	
	of thoses and charts. This waveforms depict a other shapshot of faw	
	vibration, conveying information about the state of machinery that is not	
	necessarily visible in the frequency spectrum. FFT is a technique for	
	utilising time waveform vibration signals as a vibration analysis tool.	
Fast Fourier	A time waveform is a graph or table that depicts an acceleration versus	
Transform	time. Time waveforms depict a snapshot of raw vibration over a brief	
(FFT)	period of time, conveying information about the state of the equipment that	
(III)	period of time, conveying information about the state of the equipment that	
(ER)	is not usually visible in the frequency spectrum. Using FFT as a vibration	
E	analysis tool is one approach to utilising time waveform vibration signals.	
Phase	Phase refers to the relative time difference between two signals recorded	
measurement	in angle units rather than time units when discussing vibration analysis. It	
UNIV	is effective only when the two compared signals have the same frequency.	
	Phase measurement in conjunction with FFT is used to decode machine	
	problems such as loose components, misalignment, and unbalance.	
Order analysis	Order analysis is a variant of Fourier transform analysis that is mostly used	
	to assess vibrations produced by machines with different revolutions per	
	minute (RPM). In other words, order analysis is a type of frequency	
	analysis in which the frequency axis of the spectrum is expressed in orders	
	of RPM rather than hertz. The phrase "orders" refers to a frequency that is	
	a multiple of the rotational speed of a reference object. For instance, if the	

	frequency of a vibration signal is twice that of the motor's spin, the order			
	is two.			
Power spectral	The power spectral density is calculated by multiplying the FFT's			
density (PSD)	amplitude by its various forms in order to normalise it to the frequency bin			
	width (bin width refers to the grouped x-axis values). Consider PSD as an			
	examination of "random" vibrations or motions at a variety of various			
	frequencies. PSD compares random vibration signals with varying signal			
	lengths accurately.			
Envelope	Envelope analysis is a type of vibration analysis that detects hits with			
analysis	insufficient energy, which are frequently concealed by other vibration			
and the second se	signals. It is a widely used diagnostic instrument for gear teeth and roller			
EKA				
1116	bearings that are broken.			
Orbit	The orbit is defined as a plot of the centerline of a sleeve bearing journal.			
الأك	It is determined by inserting two probes 90 degrees apart into the bearing			
LIMB	housing. The data collected by these probes may be viewed digitally and			
UNIV	utilised to detect shaft vibrations caused by oil spin - oil whirling around			
	inside the journal forcing it to move			
	inside the journal, foreing it to move.			
Resonance	In machines, resonance analysis identifies all of the inherent vibrations and			
analysis	frequencies. The existence of resonance indicates that there is a high			
	vibration present, which may reach harmful levels.			

4.3.3 Categories of Vibration Measurement

There are several categories of vibration measurement such as Overall level of vibration, Spectral analysis of vibration, Discrete frequency monitoring, Shock pulse monitoring, Kurtosis measurement, Signal averaging and Cepstrum analysis; this is explained in detail in table 4.10.

Vibration measurement	Explanation
The overall level of	Performing a "rough check" on a machine by determining the
vibration	overall degree of vibration. By feeling a machine with your hand,
AL MALAY	you can assess if it is operating in a reasonably wide frequency
EKNIK	spectrum. This preliminary inspection is most effective on rotating
L H	machinery, particularly high-speed machines. Generally, it is not
HANNO	relevant to reciprocating machinery.
the C	
Spectral analysis of	The process of converting a signal from the time domain to the
vibration UNIVERS	frequency domain is called spectral analysis. This is frequently
	accomplished through the use of FFT. The signal is analysed to
	determine the presence of any significant frequencies generated by
	the machine's components. Wherever there is a peak in the
	frequency signal, vibration is possible. Typical uses of spectral
	analysis include determining the rotational speed of a shaft or the
	frequency at which teeth mesh on a pair of gear wheels.

Table 4.10	Categories	of vibration	measurement	and its	explanation
	U				1

Discrete frequency	Consider the situation in which need to monitor a certain	
monitoring	component within a machine. In this situation, discrete frequency	
	monitoring monitors the vibration level caused by a component at	
	a certain frequency. For instance, if need to investigate a certain	
	shaft in a machine, just set the monitoring to the rotational speed	
	of that machine. The FFT algorithm is used to calculate discrete	
	frequency.	
Shock pulse monitoring	itoring Shock pulse monitoring is a preventative maintenance approach	
	that uses hand-held equipment to check rolling-element bearings.	
MALAY	The hand-held instrument emits a natural frequency when it is	
art.	struck or vibrated by rolling bearings. In other words, when two	
TEK	pieces of metal collide while in motion, shock waves are generated	
Link	and propagate through the metal. This shockwave is used to	
AIND	measure shock pulses.	
سا ملاك	اونىۋىرسىتى تىكنىكل ملىي	
Kurtosis measurement	Kurtosis is a measure of a random signal's spikiness. Signals with	
UNIVERS	a greater kurtosis value have peaks that are more than three times	
	the signal's root mean square (RMS) value. Kurtosis is used in	
	vibration analysis to monitor the development of fatigue in rolling	
	bearings using a simple instrument.	
Signal averaging	Signal averaging is critical in spectrum analysis because it	
	establishes the signal strength at each frequency. This is especially	
	critical for low-frequency data, as they require a longer average	
	period to obtain a statistically accurate spectrum estimate.	

	Averaging signals is frequently used to measure the rotational
	Trotugning signals is nequency asea to measure the rotational
	speed of gear. For instance, signal averaging will reveal the cyclic
	action of each gear tooth. If a tooth has a significant crack, it will
	be identified due to the increased flexibility of the tooth.
Cepstrum analysis	Cepstrum was originally developed to characterise seismic echoes
	generated by earthquakes and bomb blasts. It is now used to
	examine the recurring patterns in a spectrum. Repeated patterns in
	the spectrum are perceived as a single or two cepstrum components
	with several sets of sidebands, which can be perplexing. The
MALAY	cepstrum denotes the separation of those sidebands in the same
Start .	way that the spectrum denotes the separation of repeating temporal
TEK	patterns in the waveform. Cepstrum analysis is frequently used to
West	investigate the relationship between the rotational frequency of
Ainn	bladed rotors and the frequency of blade passage. Another example
سيا ملاك	is examining the frequency of gear teeth meshing and rotational
UNIVERS	gear speeds KAL MALAYSIA MELAKA

4.3.4 Vibration Analysis Measurement Parameters

These vibration analysis techniques aid in the determination of three critical parameters: acceleration, velocity (root mean square), and displacement. Each of these metrics emphasises a different frequency range and can be analysed in conjunction to diagnose difficulties. The acceleration, displacement, and velocity parameters are defined as follows:

- 1. Acceleration: Acceleration emphasises high frequencies. However, an acceleration signal is not exclusive. A signal of acceleration can be transformed to a signal of velocity or displacement.
- 2. **Displacement:** Just as acceleration emphasises high frequencies, displacement emphasises low frequencies. Generally, displacement measurements are employed to examine the wide picture of mechanical vibrations. You can utilise displacement to determine if a spinning part is unbalanced as a result of significant displacement at the rotational frequencies of the machine's shaft.
- 3. Velocity: Because velocity is proportional to the destructive force of vibration, it is the most critical parameter. It gives equal weight to high and low frequencies. Typically, the RMS value of velocity (measured between 10 and 10,000 Hz) provides the most accurate indication of vibration intensity. The RMS value is obtained by multiplying the peak amplitude by 0.707.



Figure 4.21 Graph of acceleration, displacement and velocity

Figure 4.21 above illustrates how acceleration, displacement, and velocity appear on the same signal. It is possible to notice several peaks with similar frequencies but varying amplitudes. This is an excellent illustration of how each parameter prioritises distinct frequency bands.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.3.5 Benefits of Continuous Vibration Monitoring

Not only are the methods and tools presented excellent for diagnosing what is wrong with a piece of equipment or machinery (reactive), but they may also be used to identify issues prior to them causing major downtime (proactive). Vibration analysis and monitoring enable us to investigate structural weakness or looseness statistically, as well as rotating component looseness and the presence of resonance.

Continuous vibration monitoring, when implemented appropriately, aids in the optimization of machinery operation. Modern technology enables you to take continuous vibration readings on a variety of different pieces of equipment in real-time and send the data instantly to your smartphone, tablet, or desktop through the cloud.

1. **Monitor critical equipment:** Critical equipment is any piece of equipment or machine that potentially results in a significant financial loss if it fails. Continuous vibration monitoring enables the detection of anomalies in the vibration spectrum, indicating lubrication and bearing flaws well in advance of significant failures.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2. **Monitor heavily used equipment:** Many plants operate 24 hours a day for the whole week, stopping only for periodic maintenance monthly or quarterly; halting more frequently than this might cost the plant a large amount of money. Continuous online vibration monitoring assists in monitoring the condition of heavily utilised or troublesome machinery and notifies users when the condition changes.

3. **Monitor difficult-to-access equipment:** Maintaining equipment positioned in difficultto-reach locations is challenging. Continuous vibration monitoring of rooftop machines, cooling towers, and those operating in high-temperature zones enables convenient maintenance. This avoids unscheduled downtime and prevents maintenance personnel from accessing these areas inadvertently.



4.3.6 Measurement Procedure of Vibration Analysis

The machine used is PT500.04 (Computerized Vibration Analyser), where data is taken. The Standard Operating Procedure (SOP) is shown below in table 4.1:



Table 4.11 Procedure of taking data using SENSE IT




11			After that, repeat the process but using a different value of hertz that is 15Hz and 20Hz		
12	⇒ DEGREE PROJECT (PSM) → DeepLearningE	Data	Save all the balanced data for 10		
	Name	Status Date modified			
	🖈 🛅 balance10Hz		HZ,15 HZ and 20HZ		
	balance15Hz	Ø 30/10/2021 5:57 PM			
	balance20Hz	⊘ 30/10/2021 5:57 PM			
13	UNIVERSITE		Set up machine for faulty condition by adding nut to introduce unbalance condition.		



4.4 Measurement of Data

4.4.1 Sampling rate calculation

The sampling rate or sampling frequency refers to the number of samples taken per second (or other units) from a continuous signal in order to create a discrete or digital signal. Frequencies are expressed in hertz (Hz) or cycles per second for time-domain signals such as sound waveforms (and other types of audio-visual content).

The data obtained every second in 60 seconds will be calculated and divided by 60 to get the sampling rate. The calculation of the sampling rate is as below.

MALATAIA
total number of data in each second
60 seconds
Sampling rate =
38 + 63 + 41 + 19 + 22 + 87 + 61 + 48 + 68 + 49 + 31 + 60 + 48 + 55 + 58 + 45
+30 + 62 + 67 + 42 + 32 + 74 + 54 + 39 + 42 + 47 + 58 + 57 + 48 + 46 + 38 + 60
+43 + 53 + 47 + 66 + 47 + 43 + 48 + 69 + 48 + 39 + 44 + 51 + 47 + 53 + 64 + 55
+39 + 43 + 43 + 64 + 46 + 58 + 48 + 51 + 41 + 58 + 58 + 36 + 45
60 seconds
UNIVERSITI sampling rate = $\frac{3036}{10}$ MELAKA
$60 \ seconds$
sumpting rule = 50.6

The number of data will be divided by sampling rate to get the time value, which will be plotted against acceleration.

4.4.2 Plotting the graph

The previously collected data need to be calculated to plot a graph resembling time versus acceleration, as shown below in figure 4.22.

	А	В	С	D	Е	F	G
1	AXIS	Acceleration (m/s2)	Date	Time	Number of data	time(sampling rate)	number of data(persecond)
2	X-AXIS	0.114	27/10/2021	3:41:52 PM	0	0	38
3	X-AXIS	0.287	27/10/2021	3:41:52 PM	1	0.019762846	
4	X-AXIS	0.229	27/10/2021	3:41:52 PM	2	0.039525692	
5	X-AXIS	0.057	27/10/2021	3:41:52 PM	3	0.059288538	
6	X-AXIS	0.095	27/10/2021	3:41:52 PM	4	0.079051383	
7	X-AXIS	0.114	27/10/2021	3:41:52 PM	5	0.098814229	
8	X-AXIS	0.191	27/10/2021	3:41:52 PM	6	0.118577075	
9	X-AXIS	0.057	27/10/2021	3:41:52 PM	7	0.138339921	
10	X-AXIS	0.076	27/10/2021	3:41:52 PM	8	0.158102767	
11	X-AXIS	0.114	27/10/2021	3:41:52 PM	9	0.177865613	

Figure 4.22 Snapshot figure of data analysis to plot the graph

The data obtained will be refined that is only 1 minute or 60 seconds will be selected for analysis purposes. After taking the 60-second data, the sampling rate is calculated previously, which generates the below data.

TEKNIKAL MALAYSIA MELAKA UNIVERSITI

4.5 Graphical result

Graphs can be used to visualise enormous amounts of numerical data and to depict the relationships between distinct variables' numerical values. Additionally, they can be utilised to establish quantitative correlations between variables. A graph is a graphic consisting of one or more points, lines, line segments, curves, or regions that illustrates the variation of one or more variables in contrast to the variation of one or more other variables.

A graphical display can aid in communicating and even visualising information regarding numerical data. For instance, issues concerning data patterns and whether one variable changes in response to another are frequently easier to visualise in a graph than in a table. A vibration graph depicts the behaviour of a single point along the path of a wave over time. A vibration can be thought of as a whole cycle, or back and forth motion.

For this section, the previously obtained data will be displayed visually in the form of graph which shows the relationship between acceleration data versus the time. The graph plotted will consist of 3 axis graph of X-axis, Y-axis and Z-axis. The data will be plotted for both condition which is balance condition and unbalance condition. Both condition consist of data for 10Hz , 15Hz and 20Hz which taken 3 times per data in order to apply principle of repeatability thus making the data will be turn visually into graphical form is 54 number of graph.

4.5.1 Balanced 10 hertz

4.5.1.1 Balanced 10 hertz set 1

Figure 4.23 shows the data for balanced machine condition for 10-hertz frequency set 1 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.23 Balanced 10 hertz set 1 for X, Y, and Z-Axis

4.5.1.2 Balanced 10 hertz set 2

Figure X shows the data for balanced machine condition for 10-hertz frequency set 2 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.24 Balanced 10 hertz set 2 for X, Y, and Z-Axis

4.5.1.3 Balanced 10 hertz set 3

Figure 4.25 shows the data for balanced machine condition for 10-hertz frequency set 3 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.25 Balanced 10 hertz set 3 for X, Y, and Z-Axis

4.5.2 Unbalanced 10 hertz

4.5.2.1 Unbalanced 10 hertz set 1

Figure 4.26 shows the data for unbalanced machine conditions for 10-hertz frequency set 1 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.26 Unbalanced 10 hertz set 1 for X, Y, and Z-Axis

4.5.2.2 Unbalanced 10 hertz set 2

Figure 4.27 shows the data for unbalanced machine conditions for 10-hertz frequency set 2 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.27 Unbalanced 10 hertz set 2 for X, Y, and Z-Axis

4.5.2.3 Unbalanced 10 hertz set 3

Figure 4.28 shows the data for unbalanced machine conditions for 10-hertz frequency set 3 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.28 Unbalanced 10 hertz set 3 for X, Y, and Z-Axis

4.5.3 Balanced 15 hertz

4.5.3.1 Balanced 15 hertz set 1

Figure 4.29 shows the data for balanced machine condition for 15-hertz frequency set 1 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.29 Balanced 15 hertz set 1 for X, Y, and Z-Axis

4.5.3.2 Balanced 15 hertz set 2

Figure 4.30 shows the data for balanced machine condition for 15-hertz frequency set 2 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.30 Balanced 15 hertz set 2 for X, Y, and Z-Axis

4.5.3.3 Balanced 15 hertz set 3

Figure 4.31 shows the data for balanced machine condition for 15-hertz frequency set 3 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.31 Balanced 15 hertz set 3 for X, Y, and Z-Axis

4.5.4 Unbalanced 15 hertz

4.5.4.1 Unbalanced 15 hertz set 1

Figure 4.32 shows the data for unbalanced machine conditions for 15-hertz frequency set 1 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.32 Unbalanced 15 hertz set 1 for X, Y, and Z-Axis

4.5.4.2 Unbalanced 15 hertz set 2

Figure 4.33 shows the data for unbalanced machine conditions for 15-hertz frequency set 2 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.33 Unbalanced 15 hertz set 2 for X, Y, and Z-Axis

4.5.4.3 Unbalanced 15 hertz set 3

Figure 4.34 shows the data for unbalanced machine conditions for 15-hertz frequency set 3 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.34 Unbalanced 15 hertz set 3 for X, Y, and Z-Axis

4.5.5 Balanced 20 hertz

4.5.5.1 Balanced 20 hertz set 1

Figure 4.35 shows the data for balanced machine condition for 20-hertz frequency set 1 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.35 Balanced 20 hertz set 1 for X, Y, and Z-Axis

4.5.5.2 Balanced 20 hertz set 2

Figure 4.36 shows the data for balanced machine condition for 20-hertz frequency set 2 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.36 Balanced 20 hertz set 2 for X, Y, and Z-Axis

4.5.5.3 Balanced 20 hertz set 3

Figure 4.37 shows the data for balanced machine condition for 20-hertz frequency set 3 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.37 Balanced 20 hertz set 3 for X, Y, and Z-Axis

4.5.6 Unbalanced 20 hertz

4.5.6.1 Unbalanced 20 hertz set 1

Figure 4.38 shows the data for unbalanced machine conditions for 20-hertz frequency set 1 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.38 Unbalanced 20 hertz set 1 for X, Y, and Z-Axis

4.5.6.2 Unbalanced 20 hertz set 2

Figure 4.39 shows the data for unbalanced machine conditions for 20-hertz frequency set 2 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.39 Unbalanced 20 hertz set 2 for X, Y, and Z-Axis

4.5.6.3 Unbalanced 20 hertz set 3

Figure 4.40 shows the data for unbalanced machine conditions for 20-hertz frequency set 3 for X-axis, Y-axis and Z-axis. The graph is generated from the refined data obtained during the experiment that has been done.



Figure 4.40 Unbalanced 20 hertz set 3 for X, Y, and Z-Axis

4.6 Data Validation

4.6.1 Introduction to data validation

Data validation is a critical component of every data handling process, regardless of whether collecting data in the field, analysing it, or preparing to present it to stakeholders. If data is inaccurate from the start, your findings will almost certainly be inaccurate as well. That is why data must be verified and validated before to usage.

To be more precise, data validation is a procedure that verifies the correctness and quality of your data prior to its import and processing. Additionally, it may be viewed as a type of data cleaning. Validating your data guarantees that it is full (there are no blank or null values), unique (it contains distinct values that are not repeated), and that the range of values is consistent with your expectations. Often, data validation is used in conjunction with ETL (Extract, Transform, and Load) operations, which transfer data from a source database to a destination data warehouse for analysis. Data validation enables you to confirm the accuracy of your analytical results.

While data validation is an integral aspect of any data pipeline, it is not sufficient; it is sometimes overlooked. While data validation may appear to be a phase that slows down a workflow, it is critical since it enables in order to get the finest outcomes possible. Nowadays, data validation may be accomplished much more quickly than you might believe. With data integration systems capable of incorporating and automating validation procedures, validation may be viewed as a necessary component of your workflow rather than as an afterthought.

Validating the accuracy, clarity, and details of data is necessary to mitigate any project defects. Without verifying data, the possibility of making judgments based on faulty data that is not indicative of the current situation. While evaluating data inputs and values is vital, validating

the data model itself is also critical. If the data model is not developed or constructed appropriately, difficulties will arise while attempting to use data files in a variety of applications and software.

Both the structure and content of data files will determine the extent to which data can be used. Utilizing validation rules to cleanse data prior to usage helps to reduce the likelihood of error and worst-case scenarios. Assuring the data's integrity contributes to the legitimacy of conclusions.

The data validation in this research is based on percent errors which tell how big errors happen when measuring something in an experiment. Smaller values mean that data are close to the accepted or real value. For example, a 1% error means that data got very close to the accepted value, while 45% means that data were quite a long way off from the true value. Measurement errors are mostly unavoidable because equipment can be imprecise, hands can shake, or instruments just might not have the capability to measure accurately. Percent error will let know how badly these unavoidable errors affected the results. There is no specific value of accepted error percentage because it depends on the cases.

In some cases, the measurement may be so difficult that a 10 % error or even higher may UNIVERSITITEKNIKAL MALAYSIA MELAKA be acceptable, while a 1 % error may be too high in other cases. Most high school and introductory university instructors will accept a 5 % error. But this is only a guideline at higher levels of study, and the instructors usually demand higher accuracy. This will be discussed in a detailed manner in the next section.

4.6.2 Data validation using MATLAB



Figure 4.41 Snapshot of code used in MATLAB to filter and transform the data.

For the data validation in this research, MATLAB software is being used with a code that will cutoff, filter, and transform the data obtained from the smartphone saved via Node-RED into a local machine, as shown in figure 4.41. This process will ensure and validify the data obtained from the smartphone has the same frequency as the one we set on the machine.



Figure 4.42 Time-domain to frequency-domain transform using FFT

The data obtained will be transformed from a time domain to a frequency domain, as shown in figure 4.42 above.

MATLAB® is a platform for engineers and scientists to study and build systems and products that transform our environment. The MATLAB language is at the heart of MATLAB; it is a matrix-based programming language that enables the most natural expression of computational mathematics.

There are several functions and uses for MATLAB software. The primary functions of MATLAB software are to analyse data, design algorithms, and also to generate models and applications. By deploying to enterprise applications and embedded devices and integrating with Simulink® and Model-Based Design, MATLAB enables users to take their ideas from research to production.

MATLAB is used by millions of engineers and scientists in industry and academia for a variety of applications, including deep learning and machine learning, signal processing and communications, image and video processing, control systems, test and measurement, computational finance, and computational biology.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.6.3 The result of FFT from MATLAB

4.6.3.1 The result for case 1: 10 Hertz

Figure 4.43 shows the result for case 1, which is time-domain raw data for frequency of 10 hertz and the Fast Fourier Transform (FFT) for the raw data of 10 hertz.



Figure 4.44 shows the result for case 1, which is time-domain filtered data for frequency

of 10 hertz and the Fast Fourier Transform (FFT) for the filtered data of 10 hertz.



Figure 4.44 Filtered data for 10-hertz frequency

4.6.3.2 The result for case 2: 15 Hertz

Figure 4.45 shows the result for case 2, which is time-domain raw data for a frequency of 15 hertz and the Fast Fourier Transform (FFT) for the raw data of 15 hertz.



Figure 4.46 shows the result for Case 2, which is time-domain filtered data for frequency of 15 hertz and the Fast Fourier Transform (FFT) for the filtered data of 15 hertz.



Figure 4.46 Filtered data for 15 hertz frequency

4.6.3.3 The result for case 3: 20 Hertz

Figure 4.47 shows the result for case 3, which is time-domain raw data for frequency of 20 hertz and the Fast Fourier Transform (FFT) for the raw data of 20 hertz.



Figure 4.48 shows the result for Case 3, which is time-domain filtered data for

frequency of 20 hertz and the Fast Fourier Transform (FFT) for the filtered data of 20 hertz.



Figure 4.48 Filtered data for 20-hertz frequency

From the result of FFT that being obtained it can clearly seen that the data is valid since the result of FFT and the real result of vibration is very close. The real data of case 1 is 10 Hz and obtained data shows value of 9.86, for the case 2 which is 15 Hz the FFT data obtained is also very close which is 14.87 and lastly for 20Hz which is case 3 the FFT obtained is also close which is 19.86. Thus this shows that the data set at the machine and the data captured using the SENSE IT application which make use built in acceleromter in android based smartphone is valid.



4.6.4 The Percentage of Error Calculation

For Error in % or percentage, The term "error" refers to the difference between an approximate or measured number and an exact or known value expressed as a percentage. It is a unit of measurement in science that expresses the difference between a measured or experimental value and a true or precise value.

There are a few key points to keep in mind when discussing Percentage inaccuracy. To begin, the purpose of Percentage error calculation is to ascertain how close a measured value is to its true value. Following that, for Percent or error in percentage. The term "error" refers to the difference in percentage terms between an approximate or measured number and an exact or known value.. In science, it is a unit of measurement for expressing the difference between a measured or experimental value and a real or precise value. The percent error (% error) is defined as the difference between an experimental and theoretical value divided by the theoretical value and multiplied by 100. Percent error is always stated as a positive number in some fields. In others, either a positive or negative number is acceptable. The sign may be retained to ascertain whether recorded values are regularly greater than or less than predicted values. Proceeding, one sort of error calculation is % error. Other frequently used calculations include absolute and relative error. Percent error is one component of thorough error analysis. Finally, the keys to accurately reporting % error include knowing when to eliminate the sign (positive or negative) from the calculation and when to use the right number of significant digits.

Generally, The percent error is defined as the difference between a measured or experimented value and an accepted or known value, divided by the accepted or known value and multiplied by 100%. Percent error is always expressed as a positive value in a large number of applications. The error's absolute value is divided by an accepted value and expressed as a percentage, as shown in figure 4.49.

Figure 4.49 Percentage error formula

4.6.4.1 Percentage error calculation for Case 1: 10 hertz



4.6.4.3 Percentage Error calculation for Case 3: 20 hertz AMELAKA

Percentage of Error =
$$\left| \frac{\text{accepted value} - \text{experimental value}}{\text{accepted value}} \right| \times 100\%$$

Percentage of Error = $\left| \frac{20 - 19.86}{20} \right| \times 100\%$
Percentage of Error = 0.7 %

4.7 Real-Time Warning System

4.7.1 Node-RED and Email Setup

4.7.1.1 Node-RED Introduction

Node-RED is a robust tool for developing Internet of Things (IoT) applications, with an emphasis on simplifying the 'wiring together' of code blocks to perform tasks. A visual programming environment enables developers to connect prefabricated code blocks, referred to as 'nodes,' to complete a task. When wired together, the connected nodes, which are typically a combination of input nodes, processing nodes, and output nodes, form 'flows'.

Node-RED is a programming language that enables novel and fascinating connections between physical devices, APIs, and web services. It has a browser-based editor that simplifies the process of wiring together flows by utilising a diverse palette of nodes that can be deployed to its runtime with a single click.

Initially established as an open-source project at IBM in late 2013 to address the **UNIVERSITITEKNIKAL MALAY SIA MELAKA** company's requirement for rapid connectivity of hardware and devices to web services and other applications – a sort of IoT glue – it swiftly expanded into a general-purpose IoT programming tool. Notably, Node-RED has swiftly grown in popularity and has an active developer community that contributes new nodes that enable programmers to reuse Node-RED code for a variety of purposes.

Although Node-RED was initially intended to be used with the Internet of Things, or devices that connect with and manage the physical environment, it has expanded to be useful for a variety of applications.

4.7.1.2 Features of Node-RED

The following table 4.12 below summarises the primary characteristics of Node-RED;

No	Characteristic			
1.	It provides flow editing via the browser.			
2.	It is written in Node.js and features a lightweight runtime environment as well as an event-			
	driven, non-blocking paradigm.			
3.	The numerous flows built-in Node-RED are kept in JSON format, which is readily			
	imported and exported for sharing.			
4.	Locally, it is possible to operate it (Docker support, etc.).			
5.	It fits snugly on the most popular gadgets, including the Raspberry Pi, BeagleBone Black,			
	Arduino, and Android-based devices.			
6.	It can be run in a cloud environment such as Bluemix, AWS, or Microsoft-Azure.			
	shi () (
	اويوم سيني بيڪيڪل مليسيا ملات			

 Table 4.12 Domain characteristic of node-RED

UNIVERSITI TEKNIKAL MALAYSIA MELAKA



Built on Node.js

The light-weight runtime is built on Node.js, taking full advantage of its event-driven, non-blocking model. This makes it ideal to run at the edge of the network on low-cost hardware such as the Raspberry Pi as well as in the cloud.

With over 225,000 modules in Node's package repository, it is easy to extend the range of palette nodes to add new capabilities.



Browser-based flow editing

Node-RED provides a browser-based flow editor that makes it easy to wire together flows using the wide range of nodes in the palette. Flows can be then deployed to the runtime in a singleclick.

JavaScript functions can be created within the editor using a rich text editor.

A built-in library allows you to save useful functions, templates or flows for re-use.



Social Development

20,5

The flows created in Node-RED are stored using JSON which can be easily imported and exported for sharing with others.

An online flow library allows you to share your best flows with the world.

Figure 4.50 Key features of Node-RED

Figure 4.50 above shows the key future of Node-RED, which make Node-RED the main

choice for IoT user. The main feature such as Browser-based flow editing, built on Node.js and social development.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA
4.7.1.3 Node-RED setup

Node-RED					
	0 🗅 127.0.0.1:1880,	#flow/4206fe1d5557fe75	110% 🗘		
Node-RED				- Deploy -	
۹ filter nodes	Ø Flow 1	Flow 2 ×	+ •	🔅 debug i 🖉 🏦 🗘]
~ common		Flow 2	^	🝸 all nodes 👻 🛄 all	Ŧ
		mijanjuar		"Y-AXIS,0.421,28/11/2021,7:00:05 am"	
inject		😥 set mg payload C UsersinazmeiOneDrive/Desktopix-axis cov		28/11/2021, 7:00:09 am node: 367c0114ac3643ec	
debug		😥 set mg payload 💦 C Wsern'nazmi/OreDnu/Desktoply-axis con	-	SENSE : msg.payload : string[35]	
				"Z-AXIS,10.343,28/11/2021,7:00:05 am	Ċ.,
complete				28/11/2021, 7:00:09 am node: 367c0114ac3643ec SENSE : men parload : etrine[34]	
catch 🗅	SENSE			"X-AXIS,0.095,28/11/2021,7:00:05 am"	
A. status	• connected	XAUS ()		28/11/2021, 7:00:09 am node: 367c0114ac3643ec	
		YANG M		SENSE : msg.payload : string[34]	
👌 link in 🔆				"Y-AXIS,0.344,28/11/2021,7:00:05 am"	
Ink call		Z-AUS ())		28/11/2021, 7:00:09 am node: 367c0114ac3643ec	
				"7-AXT5.10.228.28/11/2021.7:00:05 at	e.
link out		💢 move mag payload X-AXIS 👘 🔨 X-AXIS 🕑		28/11/2021. 7:00:09 am node: 367c0114ac3643ec	
💭 comment	(Y.J.2025 Y.J.2025		SENSE : msg.payload : string[30]	
	(Z-AXIS Z		"X-AXIS,0,28/11/2021,7:00:05 am"	
 function 				28/11/2021, 7:00:09 am node: 367c0114ac3643ec SENSE : msg payload : string[34]	
f function		f tripper value of email alert atatus	1 titler send emails alert	"Y-AXIS,0.363,28/11/2021,7:00:05 am"	
- switch				28/11/2021, 7:00:09 am node: 367c0114ac3643ec SENSE : msg.payload : string[34]	
27	<		× >	"Z-AXIS,9.883,28/11/2021,7:00:05 am"	1
10.0.1.1000/#43064-1-455574-75			□ - O +		

Figure 4.51 Interface of the Node-RED environment on a palette with data displayed on the side

Firstly, multiple components are available in the Node-RED environment, but only certain nodes are used for this project, mainly for visualising the data obtained and saving data. Twentyone connected nodes are being used, which comes from several categories of nodes such as common nodes, function nodes, network nodes, storage nodes, social nodes and dashboard nodes. Each node has a specific function and task assigned.

UNIVERSITI T	EKNII	KAL M	ALAYSIA	MELA	KA
Flow 2	⊘ Flow 3	Edit file node			
msg.payload		Delete		Cancel	Done
		Properties			\$
C:\Users\User\Desktop\x-axis.cs	, <u>}</u>	Filename	C:\Users\User\Deskt	op\x-axis.csv	
C:\Users\User\Desktop\y-axis.cs	· •	X Action	append to file	~	
C:\Users\User\Desktop\z-axis.cs	,		 ✓ Add newline (\n) to □ Create directory if it 	each payload? t doesn't exist?	
		Encoding	utf8	~	
		Name	Name		
		Tip: The filena to the working	ame should be an absolution of the Node-F	ute path, otherwise it w RED process.	vill be relative

Figure 4.52 Custom setting for a specific function, such as saving the file as CSV

Types of nodes	Nodes used	Image of nodes	Explanation
Common nodes	Debug	Debug	Displays the attributes of the selected
			message in the to debug sidebar tab
			and, optionally, in the runtime log. It
			displays the value of msg.payload by
			default but may be customised to
			display any property, the entire
			message, or the output of a JSON data
	MALAYS/A		expression.
P		0	
Function nodes	Function	Function	I he Function node enables JavaScript
			code execution against the messages
E			
00			that pass through it.
	AUNO .		
رك	Filter	filter on	The filter node, referred to as the
			Report by Exception (RBE) node,
UN	VERSITI T	EKNIKAL MALA	Transmits data only when the payload
			has changed. Additionally, it can
			block till a specific value changes by
			a specified amount or disregard if the
			value changes by a set amount (Dead-
			and Narrowband mode)
	Change	Change	Without resorting to a Function node,
			the Change node can be used to

Table 4.13 Nodes detail that has been used in this project

			modify message properties and set
			context properties. Each node has a
			variety of configuration options,
			including set, change, move, and
			delete.
	Switch	Switch	By evaluating a set of rules against
			each message, the Switch node
			enables messages to be routed to
			multiple flow branches.
Network nodes	MQTT in) mqtt in	Connects to the MQTT broker and
en la composition de la comp		L MA	subscribes to the specified topic's
TER	•	P	managan
E			messages.
Ctore on a day	Tril.		Adds may newload to the and of the
Storage nodes	File	file	Adds msg.payload to the end of the
رك	مليسيا ما	يكنيكل	file or replaces the existing content.
			Alternatively, it has the ability to
UNI	VERSITI T	EKNIKAL MALA	YSIA MELAKA
			delete the file.
Dashboard	Gauge	gauge	The user interface is enhanced with
		3-3-	the addition of a gauge-type widget
			the addition of a gauge type widget.
	Chart	o chart 🗹 🗖	Creates a chart from the provided
			values. This can be a time-series line
			chart, a vertical or horizontal bar
			chart, or a pie chart.

Social	Email	email	Sends the contents of msg.payload by
			email with the subject msg.topic.

After the data is transferred to the MQTT cloud server, the data is channelled to the node inside the Node-RED environment, and this is to visualise the data in a real-time manner. The data will pass through the MQTT-in node and split data received to be saved as CSV in local machine alongside with timestamp. The data also will be displayed at the side using debug node. After that, the data is visualised via dashboard using gauge and line chart. Then the data is passed through a trigger value function which will trigger the alert when an abnormal condition occurs. This will lead to an email alert status which contains information about email addresses and other important details. This then will pass a filter node which will filter and make sure to send the only true alert. Lastly, this alert will be passed to email nodes, sending an alert to the email.



Figure 4.53 Data displayed by debug node with timestamps such as date and time

Debug node shows the data and time. This makes it easier to perform analysis and keep track of data obtained as the data being obtained is abou 42 data per second, which will require a proper mangement thus, the timestamp is very important.

																		z-axis •	•	0	Q	монами	AD NAZM	IEER BIN	NAZIR AI	J MN
Fi	ile	Home	Insert	Page	Layout	Form	Fi	e I	Home	Inser	t Pa	age Lay	out	Formu	Fi	le	Home	Inse	rt Pa	ge Lay	vout F	ormulas	Data	Revie	ew Vi	ew He
Clip	board `	Font ×	Alignm v	ient Nu	% umber ~	🔛 Conc 📆 Form 👿 Cell 1	Clip	board	Font	Alig	nment	% Numb	per F	Cond Form	Clip	board •	A Font	Alig	≡ nment ×	% Numb	ber	Conditiona Format as Cell Styles St	al Format Table ~ ~ yles	ting Y	Cells	P Editing
F8	7	Ŧ	: ×	~	f _x		C94	1	•	: >	<	f _x	24	1/10/20	H9	7	Ŧ	: 2	X V	f _x						
	A		В	С		D		А		В	C	c	D)		A		В	C		D	E		F	G	Н
94	X-AXI	S	0.21 2	4/10/202	21 3:35	5:59 AM	94	Y-AXIS		0.881	24/10)/2021	3:35:5	9 AM	94	Z-AXIS		9.768	24/10/	2021	3:35:59 /	AM				
95	X-AXI	S	0.21 2	4/10/202	21 3:35	5:59 AM	95	Y-AXIS		0.881	24/10)/2021	3:35:5	9 AM	95	Z-AXIS		9.73	24/10/	2021	3:35:59	AM				
96	X-AXI	s	0.21 2	4/10/20	21 3:35	5:59 AM	96	Y-AXIS		0.881	24/10)/2021	3:35:5	9 AM	96	Z-AXIS		9.768	24/10/	2021	3:35:59 /	AM				
97	X-AXI	S	0.172 2	4/10/20	21 3:35	5:59 AM	97	Y-AXIS		0.842	24/10)/2021	3:35:5	9 AM	97	Z-AXIS		9.711	24/10/	2021	3:35:59 /	AM				
98	X-AXI	S	0.229 2	4/10/20	21 3:35	5:59 AM	98	Y-AXIS		0.861	24/10)/2021	3:35:5	9 AM	98	Z-AXIS		9.787	24/10/	2021	3:35:59 /	AM				
99	X-AXI	S	0.229 2	4/10/20	21 3:35	5:59 AM	99	Y-AXIS		0.842	24/10)/2021	3:35:5	9 AM	99	Z-AXIS		9.787	24/10/	2021	3:35:59	AM				
100	X-AXI	S	0.229 2	4/10/202	21 3:35	5:59 AM	100	Y-AXIS		0.861	24/10)/2021	3:35:5	9 AM	100	Z-AXIS		9.768	24/10/	2021	3:35:59 /	AM				
101	X-AXI	S	0.229 2	4/10/20	21 3:35	5:59 AM	101	Y-AXIS		0.881	24/10)/2021	3:35:5	9 AM	101	Z-AXIS		9.749	24/10/	2021	3:35:59 /	AM				
102	X-AXI	S	0.21 2	4/10/202	21 3:35	5:59 AM	102	Y-AXIS		0.861	24/10)/2021	3:35:5	9 AM	102	Z-AXIS		9.711	24/10/	2021	3:35:59 /	AM				

Figure 4.54 Data saved as a CSV file with a timestamp

Figure 4.54 shows the data is being saved in different files; however, with the same format, this is to ensure easy access and can be tested for each axis whenever a special or extra step is needed.



Figure 4.55 SENSE IT dashboard that display live data on gauge and line chart graph

Figure 4.55 shows the dashboard where the data is displayed visually in two different tab which is one shows a gauge, and another shows a line chart, all the data is generated in real-time.



Figure 4.56 Live data of acceleration in meter per second square on a gauge

Moving on, figure 4.56 shows the appearance of the gauge chart, which displays three different axes with different colours to resemble high, medium and low.



Figure 4.57 showed the life data of acceleration obtained in a real-time manner and plotted using a line graph.

4.7.2 Real-Time Warning Result

After all the data process and node-red setup, the result, which is a real-time monitoring alert, can be viewed via email. This means that each time abnormalities condition occurs, the system will trigger a warning which will be sent to email. The warning triggers will include time such as day, date, time, and the reading of the accelerometer, which shows abnormal value. The data from the MQTT server is passed through nodes that are set to determine any abnormal value, which is a value that is more than 10. Whenever a value that is more than 10 passes through the function, it will flag the alarm and trigger it. Then straight away, send a warning email of abnormalities condition to occur to email.



Figure 4.58 Email alert and the content of the email received each time abnormalities occur

4.8 Summary

This chapter shows the result of the data taken transformed into a signal using an FFT technique, which is very accurate as the percentage error is very low. Aside from that, an Alert warning system is also shown, which is very useful in predictive maintenance.



CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

The recent works in IoT-based condition monitoring for rotating machinery equipment have been reviewed in chapter 2. The literature review study found that most of the IoT-based condition monitoring uses vibration as a physical parameter to assess the health status of a rotating machine. Therefore, in this study, the vibration level of a rotating machine is recorded through a developed IoT platform. The design of the IoT platform has been discussed in Chapter 3, where the general methodology of the current study involves three major steps, i.e. data collection, data validation, and development of a real-time warning system. The accelerometer sensor built-in smartphone has been used to obtain acceleration data in the data collection stages. The data has been successfully transmitted to the public cloud MQTT broker in a real-time manner.

This study also uses an open-source application, namely NODE-RED, to visualise the live data and create warnings or notifications when the machine is in a faulty condition. Therefore, in the current study, two cases are evaluated. The first case is a normal operation, and the second case is a faulty operation. The developed IoT platform has been used to collect acceleration signals for both cases, and the results have been presented in Chapter 4. Moreover, the warning system has been successfully detected the faulty operation condition. Hence, the developed system and framework offers a compelling solution for faster digital visibility of critical asset such as a rotating machine.

The summary of the current work can be listed as follow:

- The application that sends data in real-time from a smartphone has been successfully developed.
- IoT platform has been successfully set up to receive and send data in a realtime manner.
- The platform to save the transmitted data and display the graphical information of transmitted data has been successfully set up.
- The warning alert system has been successfully developed with specific criteria to distinguish between the normal and abnormal conditions of the machine and sends alert emails for abnormalities that occur in real-time.
- The developed system has successfully been designed, tested, and deployed
 to perform the required task. The vibration data from a rotating machine have
 been successfully transmitted in a real-time manner, and the warning system
 has been successfully detected the faulty operation condition.
- The applicability of the developed IoT platform has been demonstrated in detecting the faulty condition of a rotating machine. In addition, the collected data has been validated using the FFT method to compare the machine's frequency with those from smartphones data.

5.2 Recommendation

Based on the current study, the following item can be implemented for future works such as:

- The application can be developed with built-in machine learning for a onedimensional conventional neural network(CNN).
- Use the Node-RED to calculate features value using Fast Fourier Transform (FFT) such as signals.
- Use Node-RED machine-learning to classify the data and train machine learning accuracy.
- Create a warning system for faulty condition detection in the application.
- Create a phyton script to do deep learning to classify the health condition of the machine.
- Save the data obtained from the alert email to do classification for the type of damage based on the value from the alert. For instance, 10 to 12 resemble a problem, most probably about a loose screw.
- Future works should use prescriptive maintenance where Artificial Intelligence (AI) is used alongside IoT for a more advanced system.
- Since the current project uses its own developed application and sends data via MQTT and is connected to node-red and later processed for normal or abnormal conditions, it is highly recommended to build one application that serves all the purposes, starting from collecting data until classifying it using machine-learning.

5.3 Project Potential

The current project has successfully designed an application that can collect the data and send it to the cloud in real-time. The IoT device used, a smartphone owned by almost everyone, can capture the machine's vibration for healthy and faulty conditions. Furthermore, the warning system has also been successfully deployed, which is used to detect the faulty operation condition of the machine. By using the developed IoT application and device and the proposed method in monitoring the health status of a machine, the industry can have better digital visibility to its vital asset. Hence, the developed framework can potentially shift conventional maintenance into a predictive and prescriptive maintenance paradigm. In addition to that, the data can be monitored. However, there is also a one-dimensional conventional neural network (CNN), a machine learning process that studies the machine data in the form of a signal and determines the health condition of the machine. The real potential is that this project can change the industry of manufacturing both locally and worldwide as it is cheap and reliable; there is vibration detection for the machine, but it is not in real-time and by using this project, the detection can be done in real-time which can prevent the machine from down or even critical damage, this would save time and obvious it will save cost and increase efficiency. The data obtained using the alert system should be collected and saved in a database that will help in future damage classification; this will make it easier to tackle the problems. The vibration analysis engineer will do this classification as the system invented is the base platform and should be further expanded with numerous potential. Aside from that, the system should also be expanded to be a prescriptive maintainance(RxM). It is a maintenance concept that collects and analyzes data about an equipment's condition to develop specialized recommendations and corresponding outcomes to reduce operational risks method. This method also combines AI with IoT. Thus, it is a very advanced technology.



REFERENCES

da Silva, A. F., Ohta, R. L., dos Santos, M. N., & Binotto, A. P. D. (2016). A Cloud-based
Architecture for the Internet of Things targeting Industrial Devices Remote Monitoring and
Control. *IFAC-PapersOnLine*, 49(30), 108–113.
https://doi.org/10.1016/j.ifacol.2016.11.137

Dias, A. L., Turcato, A. C., Sestito, G. S., Brandao, D., & Nicoletti, R. (2021). A cloudbased condition monitoring system for fault detection in rotating machines using PROFINET process data. *Computers in Industry*, *126*, 103394. https://doi.org/10.1016/j.compind.2021.103394

Goyal, D., Vanraj, Pabla, B. S., & Dhami, S. S. (2019). Non-contact sensor placement strategy for condition monitoring of rotating machine-elements. *Engineering Science and Technology, an International Journal*, 22(2), 489–501. https://doi.org/10.1016/j.jestch.2018.12.006

Souza, R. M., Nascimento, E. G. S., Miranda, U. A., Silva, W. J. D., & Lepikson, H. A. (2021). Deep learning for diagnosis and classification of faults in industrial rotating machinery. *Computers and Industrial Engineering*, *153*(September 2019), 107060. https://doi.org/10.1016/j.cie.2020.107060

Traini, E., Bruno, G., D'Antonio, G., & Lombardi, F. (2019). Machine learning framework for predictive Maintenance in milling. *IFAC-PapersOnLine*, 52(13), 177–182. https://doi.org/10.1016/j.ifacol.2019.11.172

Wu, J., & Xiao, J. (2021). Development path based on the equalisation of public services under the management mode of the Internet of Things. *Socio-Economic Planning Sciences*, *December 2020*, 101027. https://doi.org/10.1016/j.seps.2021.101027

Adikari, H. (2017). Detection of faulty sensors in IoT using Machine Learning Algorithms. May, 1–4.

Kumar, S., Lokesha, M., Kumar, K., & Srinivas, K. R. (2018). Vibration based Fault Diagnosis Techniques for Rotating Mechanical Components: Review Paper. *IOP Conference Series: Materials Science and Engineering*, 376(1). <u>https://doi.org/10.1088/1757-899X/376/1/012109</u>

Mehta, M. (2015). ESP8266 : A Breakthrough in Wireless Sensor Networks and Internet of Things. *International Journal of Electronics and Communication Engineering & Technology*, 6(8), 7–11. www.iaeme.com/IJECET/index.asp

Selcuk, S. (2017). Predictive Maintenance, its implementation and latest trends. *Proceedings* of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 231(9), 1670–1679. https://doi.org/10.1177/0954405415601640

Surendra Tambe, S. (2015). Wireless Technology in Networks. *International Journal of Scientific and Research Publications*, 5(7), 1–3.

Wieger, T., Jan, B., & Tiedo, T. (2020). Exploring predictive maintenance applications in industry. In *Journal of Quality in Maintenance Engineering: Vol. ahead-of-print* (Issue ahead-of-print). https://doi.org/10.1108/JQME-05-2020-0029

Abdulshakoor, A., Bulushi, A., Rameshkumar, G. R., & Lokesha, M. (2015). *Fault Diagnosis in Belts using Time and Frequency based Signal Processing Techniques*. 12–20.

Analog Devices. (2009). Datasheet Accelerometer ADXL335. *Datasheet*, 1–14. https://www.analog.com

Atmoko, R. A., Riantini, R., & Hasin, M. K. (2017). IoT real time data acquisition using MQTT protocol. *Journal of Physics: Conference Series*, 853(1). https://doi.org/10.1088/1742-6596/853/1/012003

Bousdekis, A., Lepenioti, K., Apostolou, D., & Mentzas, G. (2019). Decision making in predictive Maintenance: Literature review and research agenda for industry 4.0. *IFAC-PapersOnLine*, 52(13), 607–612. https://doi.org/10.1016/j.ifacol.2019.11.226

Delgado-Arredondo, P. A., Morinigo-Sotelo, D., Osornio-Rios, R. A., Avina-Cervantes, J. G., Rostro-Gonzalez, H., & Romero-Troncoso, R. de J. (2017). Methodology for fault detection in induction motors via sound and vibration signals. *Mechanical Systems and Signal Processing*, *83*, 568–589. https://doi.org/10.1016/j.ymssp.2016.06.032

Jagtap, V. A. (2017). A Review on Fault Detection of Gearbox by. 14627–14633. https://doi.org/10.15680/IJIRSET.2017.0607290

Ashton, K. (2009). That' Internet of Things' Thing. 4986.

Atzori, L., Iera, A., & Morabito, G. (2010). The Internet of Things : A survey. Computer Networks, 54(15), 2787–2805. <u>https://doi.org/10.1016/j.comnet.2010.05.010</u>

Compare, M., Baraldi, P., & Zio, E. (2019). Challenges to IoT-enabled Predictive Maintenance for Industry 4.0. *IEEE Internet of Things Journal*, *PP*(c), 1–1. <u>https://doi.org/10.1109/jiot.2019.2957029</u>

Dvali, M. F., & Belonin, M. D. (1966). Prospects for deep and ultradeep oil and gas deposits in u.s.s.r. *International Geology Review*, 8(6), 665–675. https://doi.org/10.1080/00206816609474324 Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7), 1645–1660. <u>https://doi.org/10.1016/j.future.2013.01.010</u>

Gupta, V., Simmons, D. G., & Labs, S. (2010). Building the Web of Things with Sun SPOTs.

Kwon, D., Hodkiewicz, M. R., Fan, J., Shibutani, T., & Pecht, M. G. (2016). IoT-Based Prognostics and Systems Health Management for Industrial Applications. *IEEE Access*, *4*, 3659–3670. <u>https://doi.org/10.1109/ACCESS.2016.2587754</u>

Kalpana, E., Jyothi, V., & Jyothi, V. (2020). Juni Khyat (UGC Care Group I Listed Journal) ISSN : 2278-4632 Vol-10 Issue-5 No . 5 May 2020 AIR QUALITY MONITORING USING NODE MICROCONTROLLER UNIT AND BLYNK

SOFTWARE E. KAVITHA, Copyright © 2020 Authors Juni Khyat (UGC Care Group I Listed Journal). 5, 31–35.

Lekić, M., & Gardašević, G. (2018). IoT sensor integration to Node-RED platform. 2018 17th International Symposium on INFOTEH-JAHORINA, INFOTEH 2018 - Proceedings, 2018-Janua(March), 1–5. https://doi.org/10.1109/INFOTEH.2018.8345544

Maier, A., Sharp, A., & Vagapov, Y. (2017). Comparative analysis and practical implementation of the ESP32 microcontroller module for the internet of things. 2017 *Internet Technologies and Applications, ITA 2017 - Proceedings of the 7th International Conference, September*, 143–148. https://doi.org/10.1109/ITECHA.2017.8101926

Mehta, M. (2015). ESP8266 : A Breakthrough in Wireless Sensor Networks and Internet of Things. *International Journal of Electronics and Communication Engineering & Technology*, 6(8), 7–11. <u>www.iaeme.com/IJECET/index.asp</u>

Priyana, Y., Laumal, F., & Husni, E. (2018). *Development of Earthquake Early Warning System Using ADXL335 Accelerometer. March 2018.* https://doi.org/10.31227/osf.io/sq9xr

Selcuk, S. (2017). Predictive maintenance, its implementation and latest trends. *Proceedings* of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 231(9), 1670–1679. https://doi.org/10.1177/0954405415601640

Sharma, M. P., Kantha, P., & Science, C. (2020). 'Blynk' Cloud Server based Monitoring and Control using 'NodeMCU.' 1362–1366.

Sharma, N., Shamkuwar, M., & Singh, I. (2019). The history, present and future with iot. *Intelligent Systems Reference Library*, *154*, 27–51. https://doi.org/10.1007/978-3-030-04203-5_3

Surendra Tambe, S. (2015). Wireless Technology in Networks. *International Journal of Scientific and Research Publications*, 5(7), 1–3.

ة, تىكنىكا, ملىسى

Yogendra Singh Parihar. (2019). Internet of Things and Nodemcu A review of use of Nodemcu ESP8266 in IoT products. *Journal of Emerging Technologies and Innovative Research (JETIR), June 2019*, 1085–1088.

APPENDICES



APPENDIX A The environment of coding in MIT Application



APPENDIX B Code block in MIT Application

APPENDIX C Code block in UrsAIMQTT Extension





APPENDIX D Interface of MIT developed Application

18° MQTT Websocket Client × +				-	ø ×
\leftarrow \rightarrow \mathbf{C} $\mathbf{\widehat{G}}$ \mathbf{O} $\mathbf{\widehat{G}}$ www.hivemq.com/de	mos/websocket-client/	67% 🏠	♡ ⊻ IN ©	🕒 ш 🗉 😕 🤻	⊧ 🕷 ≡
	live MQ	Websockets Client Showcase			^
	Connection Host Fost Fost Fost Fost Fost Fost Fost F	meteles BIL Clean Bession BSL Lask-Will Retain Cost			
	Publish Sub Topic Ood Retain Message	scriptions			
SAN MALAY	Messages Control Control Control				v
		ei	25°C Mostly clear	ヘ 覧 渡 (中) ENG 30/	44 AM 🖓
سيا ملاك UNIVERS	يڪنيڪل مليس ITI TEKNIKAL MALA	ر، سيتي ب ^ا YSIA ME	اوىيۇم LAKA		

APPENDIX E HiveMQ cloud server receiving data in real-time



APPENDIX F Node-RED collecting data in real-time



APPENDIX G SENSE IT dashboard for gauge meter of acceleromter

APPENDIX H Certificate of completion from MIT APP inventor



APPENDIX I Certificate of achievement from MIT APP inventor





APPENDIX J Badges and recognition from MIT APP inventor

APPENDIX K Certificate of completion from node-RED



Q http	s://discourse. nodered.org /u/Mohamad	Nazmeer/badges				${igsidential}$	± lıı\ (D 🕐 🗰
Ŵ							Q	
Summary	Activity Notifications	Messages	Badges	Preferences				
0/2 badges ma	rked as favorite							
*	Autobiographer Filled out profile information		Basic Granted all es functions	sential community	*	Certified Completed our	new user tuto	rial ப்
*	Editor First post edit	*	First Emoj Used an Emoj	ji ji in a Post ம்	*	First Like Liked a post		
*	First Mention Mentioned a user in a post	*	First Quot	te ti				
	TERUIT	LAKA		Te	N			
	يسيا ملاك	کل ما	کنید	يتي تيھ	بۇس	اون		
	UNIVERSITI	TEKNIK	CAL MA	ALAYSIA	MELA	KA		

APPENDIX L Badges and recognition from node-RED

APPENDIX M Certificate from International MAIoT Challenge 2021

