Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of ICT and Natural Sciences

Amirashkan Haghshenas

Predictive Digital Twin of Wind Farm

Master's thesis in Simulation and Visualization
Supervisor: Prof. Agus Hasan
Co-supervisor: Prof. Ottar Osen | Egil Mikalsen
June 2022

NTNU
Norwegian University of
Science and Technology

Amirashkan Haghshenas

Predictive Digital Twin of Wind Farm

Master's thesis in Simulation and Visualization
Supervisor: Prof. Agus Hasan
Co-supervisor: Prof. Ottar Osen | Egil Mikalsen
June 2022

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Department of ICT and Natural Sciences

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of
Science and Technology

Department of ICT and Natural Sciences

IE502414 - Master's Thesis in Simulation and Visualization

Predictive Digital Twin of Wind Farm

Author:
Amirashkan Haghshenas

Supervisors:
Prof. Agus Hasan
Prof. Ottar Osen
Egil Mikalsen

Company Partner:
Offshore Simulator Center

Acknowledgement

My dissertation writing has been made possible because of the support and encouragement I have received.

It is my pleasure to express my deep gratitude to my supervisor Prof. Agus Hasan who has provided me with the knowledge and understanding of the digital twin concept, which has been my most valuable asset in forming this thesis and has been guiding me to move along the right path and find my specialty. Thanks to your insight into my work, I was pushed to make better decisions and reach a higher standard.

I would like to single out Prof. Ottar Osen, my co-supervisor for his outstanding collaboration and knowledge. Your expertise and experience helped me formulate the methodology and research path. Thanks to your tremendous support and help.

In addition, I would like to appreciate my industrial supervisor, Egil Mikalsen, the CTO of Offshore Simulation Center (OSC) who enabled me to cooperate with OSC as a partner company for my master thesis and supported my research in finding the best solution and enhancing the proposal. As a result of your guidance, I was able to choose the right path and complete my master thesis successfully.

In particular, I am grateful for the unfailing support and tender care I have received throughout my life from my family. You're always supportive of me and my endeavors.

Last but not least, I'd like to thank my coworkers, friends, and classmates for their emotional and motivational support. This journey would be far more challenging without you.

Table of Contents

List of Figures

List of Tables

1 Chapter I - Introduction

Wind power is becoming increasingly popular across the world as it is playing a vital role in both renewable and emission-free energy production, making it perfect for reducing global warming and large-scale energy production. Wind turbines with modern technologies are complex machines combining aerodynamics, mechanics, and electrical with advanced control systems in order to produce energy. They continue to grow in size, and they are increasingly being deployed offshore in hostile environments with demanding situations and limited opportunities for repairs and maintenance. To ensure the systems are safe, profitable, and cost-effective, it is imperative to implement an organized operation and maintenance strategy [1].

The ongoing global digital revolution, sparked by the Industry 4.0 initiative, has brought new concepts and emerging technologies to the fore that can help these missions be accomplished. One of these new concepts is the Digital Twin, which is the digital representation of a physical asset. A digital twin is intended to accurately represent a physical object, based on data and simulation, that can be used for forecasting, monitoring, controlling and optimizing through the entire lifespan of the asset. Many applications of digital twins have already been developed, including power generation equipment, manufacturing and processes, structures, meteorology, healthcare systems, education systems, automotive industries, and city planning [2].

1.1 Motivation

The current project is an extension assignment from the last semester's specialization course in collaboration with my friend Zhicheng Hu to create an industrie 4.0 digital twin for offshore wind farm which is an effort to research the latest technologies, present and develop a digital infrastructure for wind farms. It will allow users to collect, visualize, and analyze data from the wind farms, combined with predictive capabilities and artificial intelligence methodologies which help to enhance predictive maintenance strategies, facilitate decision making, mitigate the possible failures, improve reliability and availability, and increase annual energy production.

1.1.1 Digital twin of wind farm

Wind energy is gaining traction worldwide due to several reasons, including its ability to reduce global warming, protection for ecosystems, low resource requirements, and efficiency as a renewable energy resource. The digital infrastructure for wind farms enables the collection, visualization, and analysis of wind power analytics at the asset, farm, and fleet levels. This leads to enhanced wind performance and predictive maintenance services.

1.1.2 Most common reasons of wind turbine failure

Wind turbine failures can be divided into two categories: internal and external. (Shows in figure 1) [3].

External wind turbine failures: One common cause of external wind turbine failures is damage to the blades, resulting from bird strikes, lightning strikes, rainfall, frame detachment, delamination, corrosion of the leading edge, or blade cracks.

Internal wind turbine failures: Internal failures also are divided into two: electrical and mechanical failures. There are sensitive electronic boards and systems, also essential mechanical components inside the nacelle and tower which mostly are the main reasons for internal failures and have an adverse impact on the output, performance, and reliability of wind turbines.

Electrical failures: mostly are caused by the moisture and temperature inside the converter enclosure. This environmental condition creates a seasonal conversion climate, leading to grown electrical failures that are expensive. Short circuits caused by condensation are one of the most common electrical failures. This usually happens after a scheduled or unplanned shutdown resulting in damaging the components, necessitating replacement, or at the very least, reducing the wind turbine's lifetime.

Mechanical failures: largely happen due to temperature problems, moisture reaction with metal parts that weaken and degrade mechanical elements, problems with the hydraulic and cooling system, poor alignment, balance, and calibration, blade icing, and erosion [4].

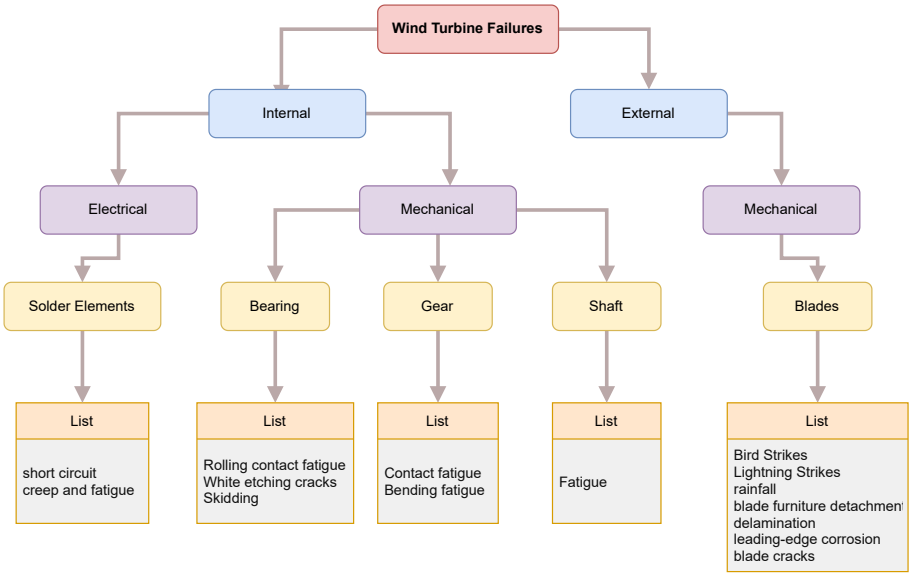

Figure 1: The figure shows different types of wind turbine possible failures divided into two internal and external sections.

1.1.3 Top 3 types of wind turbine failure

Turbine blades, generators, and gearboxes are the top three prevalent kinds of wind turbine failure. The distant locations of wind farms, weather conditions, as well as the size and height of the turbines, make it difficult to have routine maintenance and inspections also to reach the huge rotor blades and examine the blade materials and complicated surface regions during routine maintenance.

Rotor blade failures: The wind industry is looking for ways to increase the energy production of wind turbines as the demand for renewable energy develops. One way is

increasing the size of the rotor blades to boost the energy output of turbines. The rising size of blades can put more strain on the turbine's construction and other components. mechanical problems and erosion are all common defects to look out for. Lighting strikes, blade icing, material or power regulator failure, damage from external objects, and poor design are all contributing causes to blade failure which can result in costly repairs and income loss if the turbine is standstill [5].

Power generator failure: Mainly the reasons for failing in turbine generators can be wind loads, weather conditions, and thermal problems are all factors that might cause the generator to fail. The generator may fail due to manufacturing or design flaws, incorrect installation, lubricant contamination, or insufficient electrical insulation.

Bearing and gearbox failure: According to historical data and research, the bearings and gears account for 96% of the gearbox failures. Unclean lubricant, inaccurate bearing settings, temperature and vibration variations, and inappropriate maintenance are just a few of the variables that might cause failure [6].

1.1.4 Bearing failure

The goal of this project is to design and propose a predictive digital twin for wind farms to find the abnormalities and forecast the possible failures in wind turbines. Due to the vastness of the subject and the considerable variety of subtopics, it is needed to narrow the subject down and define a specific failure to work on. One of the most crucial components of rotary equipment is the bearing. The key point to maintenance effectively is accurate bearing degradation process prediction, which helps prevent failures and mitigate total maintenance costs. Therefore, bearing failures is the chosen topic for this project since they are a major source of unscheduled maintenance, repairs, and replacements, happening in the generator and gearbox resulting in energy production downtime [7].

1.1.5 Bearing most common failures

According to studies and research regarding the bearing and rolling elements failures, modes such as rolling contact fatigue, smearing, skidding, spalling, dents, and white etching cracks are the most common failures observed in wind turbine bearings most of which can be predicted and prevented by analyzing different parameters like temperature and vibration [6]. A detailed view of the bearing structure and its position within the wind turbine can be seen in figure 2 and 3.

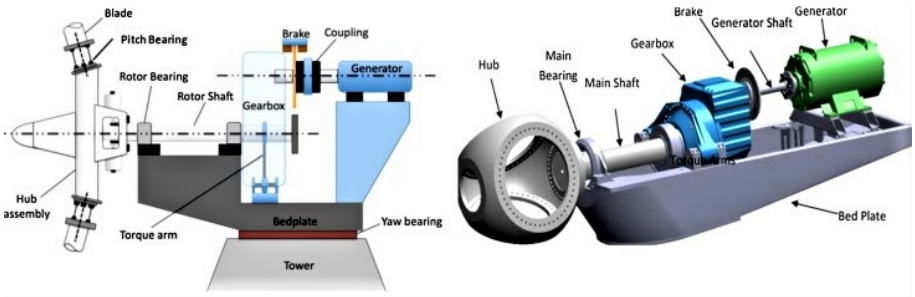

Figure 2: The structure and components of a wind turbine [8].

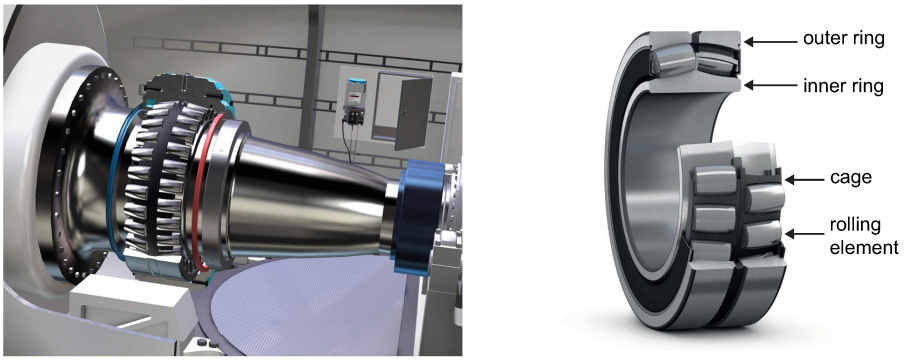

Figure 3: The Bearing structure [9]

As mentioned before wind turbine bearing is the focus of this project for finding anomalies and predicting the future condition and possible failures using the proposed solution. In the next chapter, a general discussion of predictive analytics will be presented.

1.2 Objective

The main focus of this project will be wind farms, and undoubtedly to have a robust and secure wind farm, it must be merged with a digital twin that aids the system in monitoring the functionality, facilitating decision making, and providing predictions about the asset's future behavior or evolution to avoid costly failures. This developed into the following research questions:

1. Why a predictive digital twin is a must-have item for offshore wind farms?
2. How to implement a two-way communications digital twin with prediction methods?
3. How can we predict abnormalities through a machine learning model to prevent future failures?
4. How to use the historical and real-time data for predicting the asset behavior?
5. How can we merge artificial intelligence methodologies to predictive digital twin in order to reduce the failure in future?

1.3 Literature review

This section provides an overview of the background concepts and technologies, as well as applications, existing solutions and at last wind turbine associated problems and failures. Papers, books, online courses, seminars, patents, expert interviews, and other forms of research are used in the methodology.

1.3.1 Wind farm applications

Wind power is an energy resource, and projects for wind energy are also linked to economics, project management, and other variables aside from technology. Before producing the

digital twin for it, it is needed to be familiar with the applications in this industry. Wind energy is a term that might refer to a scientific field.

According to DTU Wind Energy [10], the wind turbine industry research field is divided into three major and huge parts: Wind Energy Materials and Components(WMC), Wind Turbine Design(WTD) and Wind Turbine Design, Wind Energy Systems division (WES), and each of which has its own set of sub-sections as shown in figure 4.

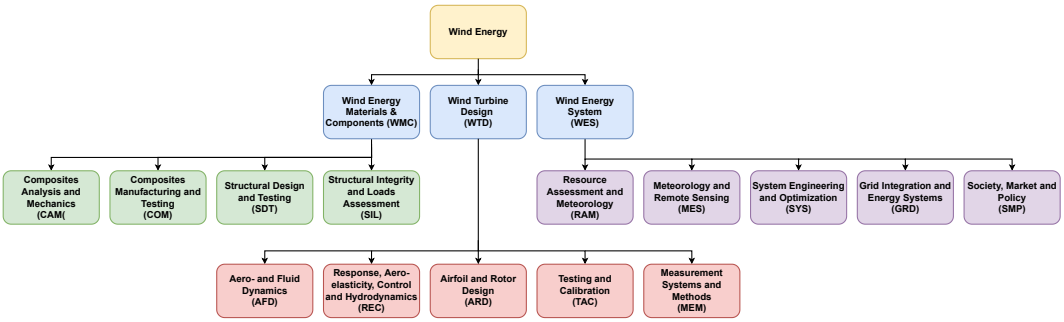

Figure 4: Windfarm energy division according to DTU university research study

The Wind Energy Materials and Components (WMC) is developing the next generation of materials, components, and structures enabling more sustainable wind turbines to be produced in large numbers to solve climate change issues. The Wind Turbine Design area covers aerodynamic design (blade and rotor design and airfoils), turbine control, and prediction of results to maximize the performance. The Wind Energy Systems (WES) section deals with sustainable wind energy, wind farm engineering, integrating renewable energy into major energy systems, resource evaluation, meteorology and remote controlling. For this sector, wind farms are a potential field and the focus of this project is on the Wind Energy Systems (WES) section including the renewable energy and offshore wind farm systems.

1.3.2 Digital Twin

Digital twin has various aspects and comes with different definitions. Boschert et al. [11] define the DT as a description of the physical and functional characteristics of a component, a product, or a system that includes more or less all information that can be useful throughout its entire lifecycle. DT is an integration of data between physical and virtual machines in either direction with ease as described by Fuller et al. [12]. A digital twin is a representation of a living or non-living object or process based on computing. By utilizing data integration, artificial intelligence, and machine learning, these models can be used to analyze current and future states, as well as simulate interventions in these objects, Knibbe et al. [13]. Glaessgen and Stargel [14] express that in order to accurately reflect the life of a flying twin, a digital twin utilizes the best available physical models, sensor updates, fleet history, etc., to create an integrated multi-physics, multiscale, probabilistic simulation. According to Verdouw et al. [15] definition, a virtual object is a digital representation of an object, with a unique identification, that can be trusted, is of integrity, is immediately available, and can serve its intended purpose.

According to all mentioned definitions, it can be said that a digital twin provides predictability, control, monitoring, and optimization of physical assets by utilizing data and simulations during the entire lifecycle of the asset and even afterward with various applications

such as health, meteorology, manufacturing, education, cities, transportation, and energy. This advanced technology comes with numerous benefits and values such as real-time remote controlling and condition monitoring, higher performance and safety, predictive and preventive maintenance, conducting what-if scenarios and risk assessment, improving decision-making systems, more useful documentation, and communication [16], [17].

Figure 5: The digital twin can be implemented for any entity ranging from industrial assets, urban cities, and even the human body and mind.

1.3.3 Digital twin typology

Digital twin has different types and is not essentially limited to an asset. They can be classified according to the magnification of the product and the application area [18]: 1-Component twins illustrate a digital representation of one component in the entire system, which are critical components with a significant impact on overall performance. 2-Asset twins are a combination of two or more components that work together describing how they function together. 3-System twins make it possible to combine individual assets into a functioning system. 4-Process twin represents entire production plants and shows how systems operate together 5-Network twin depicts a simulation model of a communication network that includes the operational environment and the application traffic.

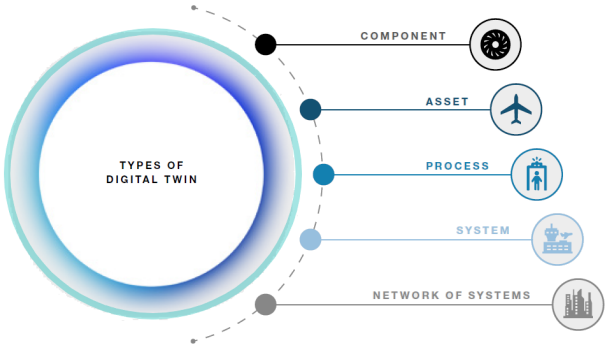

Figure 6: Different type of digital twin from small component to a huge system and network.

According to Verdouw et al. [19] literature review, a digital twin is defined by six typologies:

Virtual twins: are conceptual entities that represent objects that do not yet exist physically. IIn other words, it represents the information that needs to be fabricated in order to realize the physical twin for example functional requirements, 3D product models and materials, production models, and disposal and recycling specifications [15].

Monitoring Digital Twins: tracking the state, behavior, and trajectory of a real-world physical object. It monitors its condition and operation in real-time and links real-time to its physical twin. Monitoring Digital Twins can both provide insight into what is happening with the connected objects and also provide diagnostic information about why things occur, by linking the object to contextual data.

Predictive Digital Twin: An interactive digital twin that predicts the future states and behavior of objects based on predictive analytics, which involves statistical forecasting, simulation, and machine learning methods. Physical twin data are used to make dynamic predictions based on real-time information. An interactive digital twin that predicts the future states and behavior of objects based on predictive analytics, which involves statistical forecasting, simulation, and machine learning methods. Physical twin data are used to make dynamic predictions based on real-time information.

Prescriptive Digital Twin: a smart digital item with intelligence for advising remedial and prohibitive measures on real-world objects, often using heuristics and optimization algorithms. Prescriptive Digital Twin: a smart digital item with intelligence for advising remedial and prohibitive measures on real-world objects, often using heuristics and optimization algorithms. The input for prescriptive twins will be the outcome of predictive and monitoring twins to provide decision options and measures that are needed to be taken by humans to accomplish the proper result [20].

Autonomous Digital Twin: Without human interaction on the field or remotely, it runs independently and fully controls the behavior of real-life entities. They can also be evolved into flexible systems that can be adapted and learn about their surroundings, diagnose their own service requirements, and adjust to the priorities of users [21], [22].

Recollection Digital Twin: Keeps track of the whole history of a physical thing that no longer exists in the actual world. This form of Digital Twin is becoming increasingly significant for lowering the environmental effect of disposals and optimizing next-generation items [23].

1.3.4 The advantages of digital twin

Digital twin brings valuable benefits that some of them are mentioning here based on the report from Oracle [16], [2]:
1- Real-time remote control and monitoring: It is nearly impractical to physically obtain the detail of a large system in real-time. However, by using a digital twin the system performance can be observed easily along with remotely controlled.
2- Higher safety and efficiency: Digital twins bring more autonomy to the industry and systems which means less human interference by controlling remotely and drastically decreasing possible dangerous work conditions for humans resulting in focusing more on innovation and creation.
3- Predictive and preventive maintenance: By Collecting real-time data accurately via the physical sensors in a digital twin, a robust predictive system can be implemented by

applying intelligent data analysts to fault diagnosis in advance resulting in providing a proper maintenance schedule.
4- What-if scenarios: Digital twinning enables what-if scenarios and analysis which provide decent risk assessment and a suitable platform for testing and management away from imperiling the real asset and additional financial expenses.
5- well-organized decision system: Real-time data analysis of the system through a digital twin allows to have an efficient decision making and support system.
6- Improvements in documentation and communication: Digital twin enables stakeholders to keep well informed by providing real-time, readily available information.

Figure 7: The importance of the digital twin in state of the art technologies.

1.3.5 Applications of digital twin

There are diverse applications of digital twin proposed and are being used which have been discussed in [2] that will be briefly described as follow.

Energy: In the energy sector, digital twins have developed as a revolutionary technology that allows for the design, monitoring, and prediction of asset performance. The authors of [24] examined how to integrate digital shift technology into smart factories to enhance energy efficiency and reduce both production expenses and carbon gas emissions. [25] towards enhanced Building Information Modeling (BIM) systems addresses fault identification and diagnostics challenges in the constructed environment. The authors presented BIM digital twin for improved examination and evolution of residential building design, structure, and performance in [26].

Smart cities and transportation: Digital twin technologies are growing every day providing more advanced and smart solutions to support the decision-making process in urban development-related tasks such as monitoring and controlling transportation, walkability, air pollution, traffic congestion, and consequently making decisions according to the acquired data. In [27] author has explained a smart city model providing the capability of enhancing insight into the interaction of the human-based city technologies, where spatiotemporal urban data and properties merged into an analytical platform in real-time. [28] suggested a cyber-physical system method for tackling large-scale road shipment transportation concerns, such as decreased fuel utilization and integrated road and traffic planning. In [29], the authors developed a cross artifact explanation for air traffic management that includes physical interaction and augmented reality. The authors of [30] investigated a digital twin concept to employ residents in city planning by connecting real things with digital equivalents.

Meteorology: The concept of the digital twin broadly is being used in all meteorological institutes applying physics simulators, terrain models, radars, sensors, satellites, and other data resources to provide weather forecasting in the short and long terms. Within the field of meteorology, digital twin technologies have been so successful because the privacy and confidentiality of personal data have been lower than in other fields [2].

Manufacturing and industries: Industries are the biggest exploiters of the digital twin concept. By emerging the Industrie 4.0 standard offering technologies such as industrial IoT, augmented reality, cloud services, cybersecurity, robotics, etc, the digital twin is being employed in order to improve the efficiency in production, safety, and cost reduction. [31] conducted a systematic study on digital twin applications in industrial environments, finding that the most widespread application area is the advanced prognostics and health management systems. To facilitate the model-based design and digital twin applications between suppliers and manufacturers, a functional mock-up interface (FMI) has been proposed as a standard independent of any tools [32], [33].

Education: The digital twin has brought valuable opportunities and technologies in personalizing education with more efficient ways of teaching through online courses, counseling, artificial intelligence controlling bots for evaluations, and giving assignments making it much more comfortable and impressive to learn educational content. The authors of [34] exhibited a digital data administration system for authorization purposes in order to aid the evaluation process by taking data collection, description, investigation, and implementation into consideration.

Health: The advent of smart wearable gadgets, the need for personalized and targeted medication, and the combination of engineering and medical disciplines provide the context for the development of smart health and exploiting most of the benefits of the digital twin. It ranges from remote surgery using the digital twin framework discussed in [35], training before conducting the actual surgery, tissue engineering for more predictable, dependable, and beneficial clinical results suggested by [36], organ on a chip development overviewed by [37], psychological systems, etc, are examples of application in the healthcare sector.

1.4 Contributions of this thesis

The main contributions of this thesis are:

1. Implementing a bi-directional digital twin platform concluding physical components and digital 2D-3D and AR visualization.
2. An approach to combining data-driven models operating on the virtual twin to predict the behavior of the physical asset in real-time.
3. Applying and evaluating different prediction machine learning models to forecast time-series data
4. Modeling and estimating anomalies in the vibration and temperature evolution of wind turbine bearings over long time periods using these methods.
5. Integration of insights generated by the predictive twin into operations and processes using artificial intelligence.

1.5 Outline of this thesis

In the first chapter, technology definitions were discussed, a literature review about the most common failures in wind turbines, and contributions. The second chapter will be about general ideas for predictive digital twins and how to predict a failure in a system. In the third chapter, the topic will be narrowed down to a specific example of turbine failure and followed by chapter four which is about the proposed solution and the effort for integration, implementation, and visualization in the Unity3D platform. Chapter five is followed by the evaluation of the simulation results and discussions, and finally, the conclusions with reflection and recommendations in the last chapter.

2 Chapter II - Predictive Digital Twin

2.1 How to predict failure

It is crystal clear how a product breakdown and failure can impact original equipment manufacturers in terms of losing reputations, production, interest, and profit opportunities along with a lot of losing time and money. Applying predictive methodologies to estimate the failure probability gives the ability to schedule on-time maintenance for reducing repair time and unplanned maintenance, as well as organize proper spare parts to mitigate inventory costs and raise the bottom line. The failure prediction process can be performed in steps:

1. recognizing the main problem
2. providing a proper infrastructure to collect the required data using different methods for building and extracting the useful data
3. cleaning, preprocessing and visualizing the data
4. applying models to test and deploy a decent result to achieve a steady behavior trend.
5. finding the possible future anomalies and failures.
6. decision making and problem-solving.

Mostly to have a prediction failure process in automation and manufacturing industries, real-time data is collected by utilizing supervisory control and data acquisition (SCADA). The acquired data is cleaned and processed by modern procedures and technologies which is straightforward to do with Python libraries and other available software. For the modeling part, different methods like machine learning and deep learning techniques such as ARIMA, Regression methods, LSTM, SVM, etc, are applied to build the proper and accurate model of the system and prepare it for forecasting and finding the expected trends. Finding the outliers and abnormalities based on the behavior forecast and at last decision making to avoid potential failures.

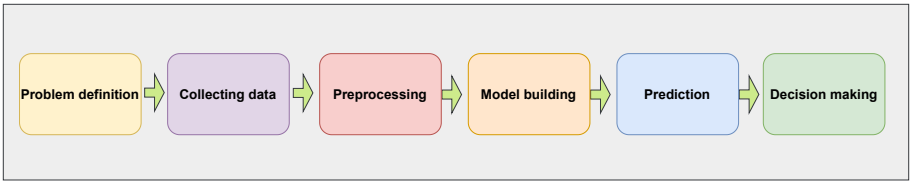

Figure 8: General failure prediction procedure.

2.1.1 Prediction methodologies

The prediction of wind turbines is mostly conducted based on the time scales of short-, medium-, and long-term. Long-term prediction focuses on estimating the variables over time periods of days, whereas medium-term prediction examines the variables at hourly intervals and Short-term forecasting seeks to estimate the values at 10-second or 10-minute intervals. For example, Wind turbines are controlled through short-term prediction; medium-term prediction is used to determine generation mix and power system maintenance schedules, and long-term prediction is used for specifying generation combination and maintenance plans.

Different prediction methods are used to forecast the wind turbine condition which can be divided into four categories of statistical models, physics-based methods, data mining algorithms, and hybrid models [38].

Statistical forecasting approaches are popular because of their objectivity in analyzing data and identifying patterns that can be used for future forecasts [39]. The most common models used in this method are Simple moving average (SMA) providing quick and rough estimates, different variations of the autoregressive (AR) methods such as Autoregressive integration moving average (ARIMA) used for time series data, Neural network (NN) which offers more precise predictions with simpler algorithms.

Physics-based methodologies are based on the physical models that define and represent the behavior of the variables providing more reliable predictions for the future trend of the model.

In general, the nonlinear part of the data is overlooked as numerical methods concentrate on capturing the linear data. Data mining methodology is used to model both linearity and non-linearity of the data. Data mining models are created through heuristics and calculations. It first looks for patterns or trends among the provided data to create a model. neural network (NN), support vector machine (SVM), K-nearest neighbor (KNN), and tree-based regression algorithms are examples of data mining methods used for prediction [38].

In this project, data-driven prediction algorithms have been chosen and applied to conduct experiments and achieve high accuracy of forecasting in different time scales. Methods applied and examined for this project are LSTM, SVM, Arima, Auto-Encoder Neural Network, and LightGBM developed by Microsoft, and Prophet software developed by Facebook.

2.1.1.1 Long-short-term memories (LSTM)

LSTM is used in recurrent neural networks to learn to forecast future events based on sequences of variable length. LSTM is used in recurrent neural networks to learn to forecast future events based on sequences of variable length. The primary principle underlying LSTM cells is to remember the most significant elements of the sequence while forgetting the less important ones. In contrast to single data points such as images, LSTMs have feedback connections, which means they are capable of processing an entire sequence of data. It has useful applications such as Language modeling, Handwriting recognition, speech recognition, etc [40].

2.1.1.2 Support Vector Machine (SVM)

SVM is a prominent Supervised Learning technique that is utilized to solve both classification and regression issues. However, it is mostly applied in Machine Learning for classification problems. This algorithm aims to construct the best line or decision boundary in n-dimensional space that can be used to categorize the data easily in the future, which is known as a hyperplane, to easily place the new points in the correct category. SVM creates the hyperplane by choosing extreme points. As a result, extremes are named support vectors, and the algorithm that employs them is known as a Support vector machine. [41]

2.1.1.3 AutoRegressive Integrated Moving Average (ARIMA)

ARIMA is a statistical model that uses previous values to describe the values of a time series. This model is built on two basic characteristics of past values and past errors. When the model is built, past values or the last time-series data are considered features, which are hyperparameters to be determined. The model can utilize the historical data to determine the performance of the past by using past errors [42].

2.1.1.4 Auto-Encoder Neural Network (AENN)

Autoencoders are a type of unsupervised ANN (Artificial Neural Networks) learning the compression and encoding the data efficiently before learning to rebuild and decode the data from the compact encoded model into a representation close to the original input as likely. It is basically designed to minimize data dimensionality by learning how to disregard data noise [43].

2.1.1.5 Light Gradient Boosting Machine (LightGBM)

The LightGBM open-source framework is a Gradient Boosting Decision Tree (GBDT) developed by Microsoft to create an efficient and distributed tree-based learning algorithm. With LightGBM, large datasets can be trained with minimal memory overhead, which is highly efficient. It can more efficiently handle high-dimensional sparse features as well as train trees with a small fraction of the full dataset [44].

2.1.1.6 Prophet Forecasting Tool

Prophet is an additive model that fits non-linear trends with annual, weekly, and daily seasonality, as well as holiday impacts, to forecast time series data. It is an open-source software developed by Facebook's Core Data Science team. It is used for time series forecasting with substantial seasonal consequences and chronological data from several seasons. Generally, Prophet can handle outliers and missing data well and is robust to changes in the trend.

This tool has been chosen as the prediction solution for this project, therefore, it will be discussed in more detail as follow:

Prophet is a time series prediction prototype developed to manage the typical features of business time series. Moreover, it includes intuitive parameters that can be adjusted without any knowledge of the underlying model. It has been designed using three-part decomposable time series models of trend, seasonality, and holidays. There are different prediction methods and each method will address a specific predicting issue. Prophet is designed for the types of business forecasting activities, which often include one or more of the following: [45]

- non-linear growth curves, when a trend reaches its natural limit or saturates
- strong multiple "human-scale" seasonalities: day of week and time of year
- important holidays with established dates that occur at unpredictable intervals
- a sufficient number of missing observations or significant outliers
- trend shifts in the past, such as product releases or recording changes
- non-linear growth curves, when a trend reaches its natural limit or saturates

All of which makes this software an ideal prediction solution for maintenance operations and failure forecasting.

The model is defined as the following equation Eq.1 [46]:

$$y(t) = g(t) + s(t) + h(t) + \epsilon(t) \tag{1}$$

Where:

g(t) reflects non-periodic variations in the value of the time series.

s(t) indicates periodic variations such as weekly and yearly seasonality.

h(t) refers to the influences of holidays that coincide with possible irregular schedules that overlap one or more days.

The error term ϵ(t) is used to refer to any characteristic modifications which are not captured by the model.

A key component of generating data for growth forecasting is a model of the population growth and how it is predicted to continue.

The Growth Function

There are three alternatives for the growth function:

Linear Growth: Prophet's default option is linear growth. It employs a collection of symplectic linear equations with diverse slopes at variation points. When linear growth is implemented, the growth term will look like the classic equation y = mx + b; however, the slope(m) and offset(b) will be changeable and change value at each successive changepoint.

Logistic Growth: Useful when modeling time series with a top limit or a floor under which the values you are estimating have a maximum or a minimum value and cannot exceed those values. The growth term for logistic growth looks similar to a typical logistic curve equation (see below), except it will have a varying carrying capacity (C), a variable growth rate (k), and an adjustable offset (m) that will change at each change point. The equation will be presented as below Eq.2 [47].

$$g(t) = \frac{C}{1 + \exp^{-k(t-m)}} \tag{2}$$

Flat: If there is no growth over time (yet there may still be seasonal variation), then the trend can be chosen as flat. This will produce a constant growth function.

The Seasonality Function

Based on the needs of the data set, the seasonality function is simply a Fourier Series as a function of time which is measured by the following equation Eq.3 [47].

$$s(t) = \sum_{n=1}^{N} (a_n \cos(\frac{2\pi nt}{p}) + b_n \sin(\frac{2\pi nt}{p})) \tag{3}$$

It is also possible to choose between additive and multiplicative seasonality, as the higher the order the more terms there are in the series.

The Holiday Function

Prophet has a holiday function that allows it to adjust forecasting when there is a holiday or a major event. The algorithm takes a list of dates and adds or subtracts based on historical data on the analyzed dates from the forecast, as a function of the growth and seasonality terms.

2.1.2 Predictive digital twin

A predictive digital twin of wind farm uses the edges of the digital twin to offer a variety of benefits including advanced wake models and farm control, leading blade edge erosion, degradation of bearings and gears, fatigue monitoring, cable health monitoring, blade damage monitoring and power electronics health monitoring. [48] Preventive maintenance is one technique to lessen the likelihood of a wind turbine failing and extending its lifespan. Condition monitoring of temperatures, vibration signatures, and component structural

integrity can all be used to predict potential breakdowns. Understanding the fundamental cause of various forms of failure can lead to design improvements and increased component reliability. Additionally, developing dependability models for various components aids in risk management and maintenance planning and avoiding costly standstills [5].

2.1.3 Digital twin of wind farm

Digital twins of wind farms are beneficial for monitoring and operating wind turbines remotely. They enable effortless and cost-effective maintenance and optimization and ensure greater reliability of the components used to convert wind energy into electricity. Oñederra et al. [49] outlines using a digital twin for a medium voltage cable prototype in a wind farm which can be used to simulate its behavior and lifespan in order to accomplish preventative maintenance. A hybrid model of a dynamic medium voltage cable model and an interpolation technique was created in the OpenModelica. As part of a predictive maintenance plan, [50] provides a unique approach for predicting the remaining useful life (RUL) of an offshore wind turbine in a digital twin shell for monitoring the turbine conditions. Wang et al. [51] summarize recent work in the reliability investigation of offshore wind turbine support structures and review the probable damages and embedding of digital twin technology into the offshore wind turbine support structures as a solution to problematic challenges. [52] conducted research to apply digital twin frame to use the gathered vital data from attentively chosen spots of hybrid wind turbine structure for updating material models of the wind turbine in order to improve maintenance and operating parameters and extend the turbine's useful life. in article [53] the author represents a summary of recent research on modeling methodologies to build a digital twin for wind turbine by considering the components, aerodynamics, structural and mechanics, power electronic converters, pitch and yaw systems. Pimenta et al. [54] applied a digital twin by using SCADA to create a feasible trustworthy numerical model of a floating wind turbine in order to reproduce experimental data for testing monitoring instruments on these wind turbines.

2.2 Exist Software solutions for predictive digital twin

Regarding predictive digital twin, a variety of research and study has been conducted and some methodologies have been proposed to utilize physics-based, data-driven, or hybrid models and digital twin concepts, in order to facilitate predictive and preventive maintenance using Artificial Intelligence, Machine Learning, Kalman Filter, Observers, etc.

As of right now, there are various solutions delivering benefits and features associated with digital twins for wind turbines and wind farms like Offshore Simulation Center (OSC) and its daughter company AugmentCity providing digital twin, virtual prototyping and prediction tools for maritime operations and city simulation, Kognitwin Energy providing visualization, simulations and physics-based models enabling field validation of measurements, as well as solutions that are utilized for the wind turbine or wind condition simulation such as DNV GL, AirSim, OpenFast, ASHES, etc using the offline data for simulation and forecasting. Several well-known wind turbine manufacturers, such as GE, Siemens, and Vestas offer platforms that are specifically used for wind farms and will be debated in more detail.

The first Software solution is "Scipher" designed by Utopus Insights using by energy companies like Vestas company which is one of the world's largest suppliers of wind turbines.

Scipher is an energy analytics platform using different cores for visualization, performance, forecasting, and maintenance. Scriper.Vx core used for a ringside view of energy assets offers portfolio-wide asset visualization, Scipher.Vx+ offers proactive asset performance monitoring analytics and the Scriper.Fx core is designed to provide forecasting of renewable energy sources, enabling the system to be more predictable and dependable [55], [56].

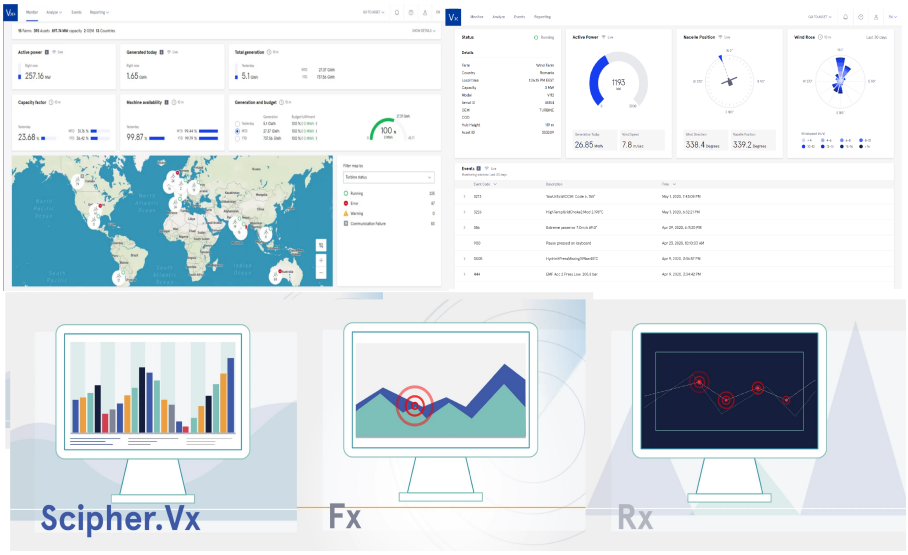

Figure 9: Scipher software solution by Utopus Insights used by Vestas company. [56]

Another solution is CMS X-Tools designed and used by the advanced company Siemens for condition Monitoring and MindSphere, the cloud-based, open IoT operating system which turns the condition data into digital added value throughout the entire asset [57], [58].

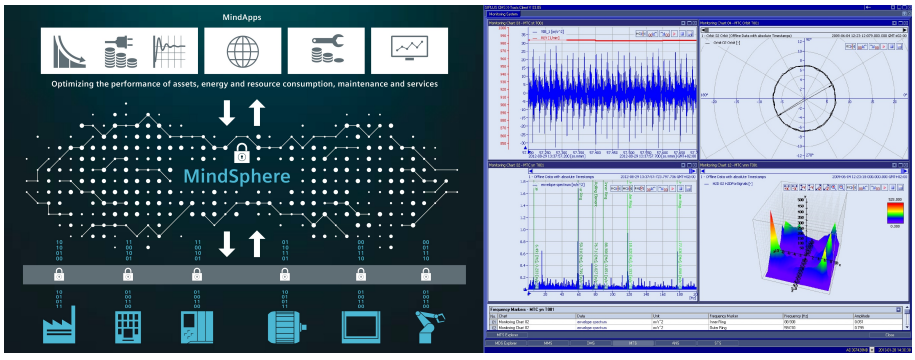

Figure 10: CMS X-Tools and Mindsphere software solution by Siemens [59].

General Electric, which is also one of the leading suppliers of wind turbines in the world, uses its own Predix platform, which is an industrial IoT software platform for edge-to-cloud data connection, analytics, and services to support industrial applications. It provides real-time visibility, communications, and control by using software and applications such as WindSCADA, Edge to Cloud, eSCADA, and Cybersecurity Services and it is more likely an intermediate product between traditional SAP, MES, SCADA, IoT, OPCUA and fully digital twin.

Figure 11: Predix software solution by General Electric.. [60]

3 Chapter III - Bearing Failure Prediction

Since the project focuses on predicting the failure of wind turbine bearings, thus, health diagnosis, anomaly detection, and bearing's behavior prediction are needed to prevent the failures in the future.

Several methods have been developed for assessing the operating conditions of the wind turbine bearings, which can be divided into two categories: vibration-based and temperature-based observations.

3.1 Bearing health monitoring

Bearing health status can be measured and monitored based on different factors and parameters, however, temperature and vibration are the two most notable indicators of whether bearings are functioning normally or abnormally. Various methodologies are used for evaluating the rolling element remaining useful life (RUL). Currently, technologies such as machine learning and artificial intelligence lead to the most accurate and efficient methods of predicting and are being used to monitor and predict the asset's condition and behavior.

3.1.1 Bearing vibration parameter

Vibration monitoring and analysis in rotating elements provide critical information concerning anomalies produced inside the machinery's internal structure, as well as the ability to plan maintenance actions [61]. Vibration measurement and interpretation are both parts of the vibration analysis approach. The information gained from the vibration signals is used to predict destructive failures, increase asset usage and efficiency, extend the life of assets, and lower maintenance costs related to asset health [62]. As long as a machine is in good condition, its vibration spectrum is modest and steady. When problems occur and parts of the machine's dynamic processes change, the vibration spectrum changes as well.

3.1.2 Bearing temperature parameter

There are potential high or low-temperature issues that result in failures in bearing. High temperature leads to a decrease the lubricant viscosity, loss of lubricant, reduction of the lubricant life, fatigue life, seals drying and cracking, changing in fits, and weakening retainer and cage. Low temperature causes raise the lubricant viscosity, skidding, increasing

torque, Lubricant viscosity becomes too high, leading to increased torque, seal failure skidding, etc. [63] Data on the temperature time series of the wind turbine bearing have a firm correlation with their historical values as well as with other related external variables, like active power, gearbox bearing and oil temperature, hub temperature, ambient temperature, rotor RPM, etc. Therefore, the prediction of temperature changes is essential for overheating warnings [64].

3.2 Vibration and Temperature analysis methods

The vibration signals are first gathered in the time domain using a vibration sensor, then applying some vibration processing methods and factors to extract the useful information from vibration signals in order to forecast the potential failures. It will be explored and discussed several trend methodologies and functions for vibration analysis that are going to be used in the proposed solution.

3.2.1 FFT (Fast Fourier Transform):

FFT is a vibration waveform-based technique of analysis to change a signal from the time domain into the frequency domain. Waveforms are complex and hard to examine in general and the FFT method helps to break down the waveforms into a succession of distinct sine waves or assess them separately. A signal is converted from the time domain to the frequency domain using FFT [65].

The FFT method can be used to detect vibrations produced by a machine, for example, when it is operating abnormally due to misalignment or bearing detriment, and shows outlier signals as shown in figure 12

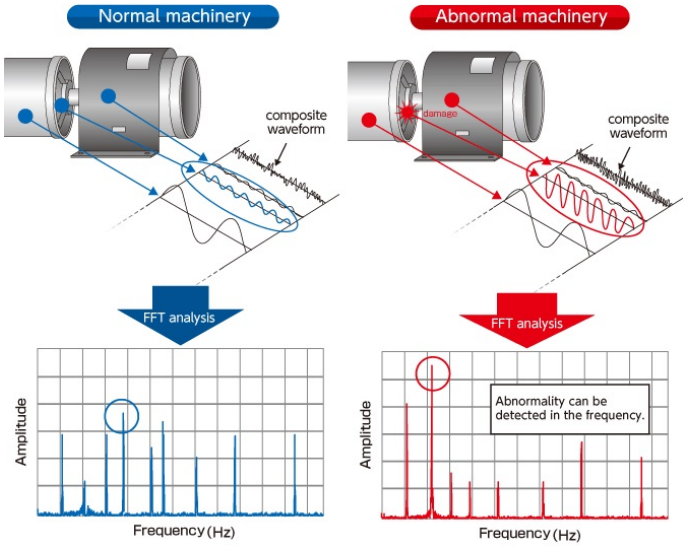

Figure 12: Example of showing abnormalities using the FFT method. The left figure shows the FFT chart during normal operation of a motor and the right one shows the FFT chart during abnormal operation [65].

There are four possible bearing failing frequencies defined as follows [66]:

Ball Pass Frequency Outer (BPFO): Measure of how many balls pass through an outer ring during an entire rotation.

Ball Pass Frequency Inner (BPFI): Indicates the number of balls passing through a given point in the inner ring over the course of a full rotation.

Ball Spin Frequency (BSF): The number of times a roller will turn during a full rotation.

Fundamental Train Frequency (FTF): The number of rotations a bearing cage makes during a full rotation of a shaft.

Example of the process of using mentioned frequencies to predict failure can be shown in figure 13.

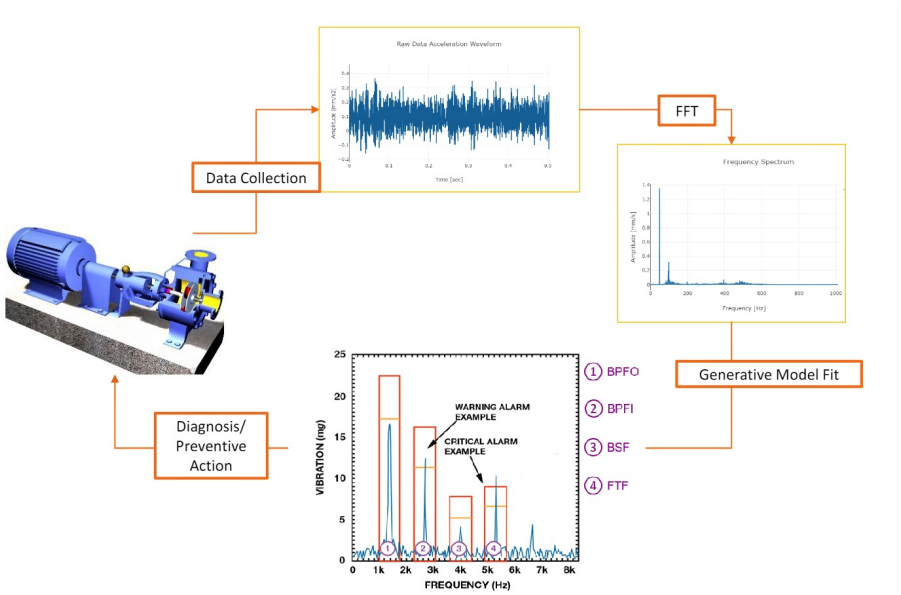

Figure 13: Physical model of using FFT for diagnosis and preventive actions. It starts with data collection by sensors, data processing, using FFT method, and finally finding the abnormalities [66].

3.2.2 RMS (Root Mean Square Value):

A vibration profile's Root Mean Square value is linked to its energy content and, therefore, to its destructive capacity. As such, the RMS of bearing vibration signals is often regarded as an indicator of bearing life, since it helps reveal comprehensively and identically the remaining life of the bearing. In most cases, the RMS (root mean square) number is the most informative. It is calculated by taking the square root of the average of the squared value of amplitude (Eq.4) [65], [66].

$$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N} X_i^2} \tag{4}$$

where N is the number of discrete points and Xi denotes the signal value at each point.

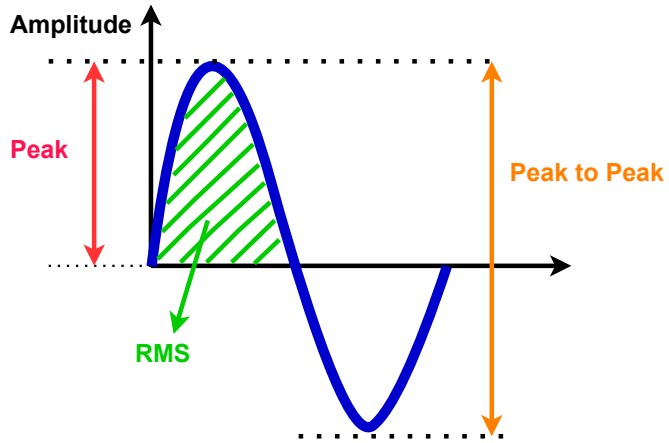

Figure 14: the RMS value of the sine wave equals the area beneath the half-wave.

3.2.3 Kurtosis:

Kurtosis describes the behavior of the tails. It is, in fact, the number of outliers in the distribution. Kurtosis is used in vibration or temperature analysis to track the progression of fatigue in rolling bearings and indicting the running state [67]. It has three types described as [68]:

Mesokurtic: means that the tails of the distribution are equivalent to the normal distribution. For the normal distribution, the kurtosis is 3.

Leptokurtic: If the kurtosis is larger than 3, it is considered leptokurtic. The tails will be weightier than the average distribution in this example, indicating that the data contains many outliers. It has a narrow bell-shaped distribution with a higher peak than the typical distribution.

Platykurtic: Kurtosis is less than 3, indicating a narrower tail or fewer outliers than a regular distribution. The bell-shaped distribution will be wider and the peak will be lower than the mesokurtic.

The numerical value is calculated by the following equation Eq5

$$Kurt = \frac{1}{N}\sum_{i=1}^{N}(\frac{X_i - m}{\sigma})^4 \tag{5}$$

where Xi is ith variable of the distribution, m is mean of the distribution and N is number of variables in the distribution.

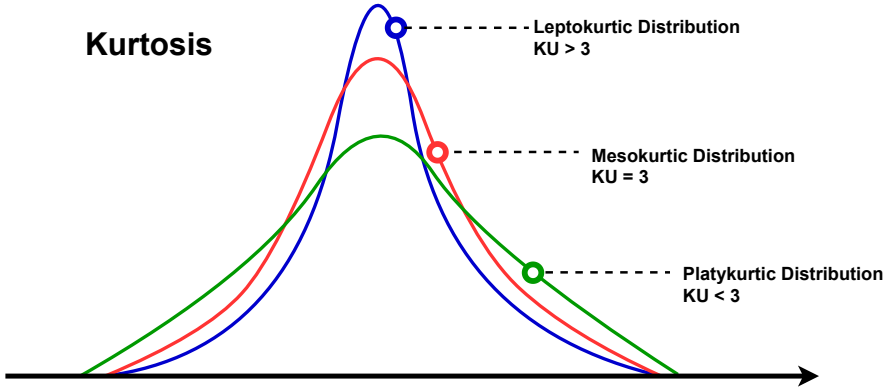

Figure 15: Three different types of Kurtosis value

3.2.4 Skewness:

The lack of symmetry in distribution is described as skewness. The Mean, Median, and Mode are all the same in symmetrical distribution and the skewness of the normal distribution is zero. Skewness provides information about the distribution of the data. It has two types [68]:

Positive skewness is defined as a longer or wider tail on the right side of the distribution.

$$Mean > median > mode$$

Negative skewness is described as the tail on the left side of the distribution being longer or wider.

$$mean < Median < mode$$

Consider bearing temperature ranging from 0 to 100 ℃ with the average being 50. When the peak of the distribution is on the left side, that means that the data is positively skewed, and most of the time, bearings are working at temperatures lower than average.

When a peak appears on the right side, data is negatively skewed, which means the bearings are typically working at higher temperatures than normal.

It can be considered that the data to be fairly symmetrical if its skewness is between -0.5 to +0.5, moderately skewed if its skewness is between -1 to -0.5 and heavily skewed if its skewness is less than -1 and greater than +1.

This factor can be used for finding anomalies and outliers in data distribution, like vibration or temperature of bearings. The values is measured by the following equation Eq.6:

$$Skew = \frac{1}{N}\sum_{i=1}^{N}(\frac{X_i - m}{\sigma})^3 \tag{6}$$

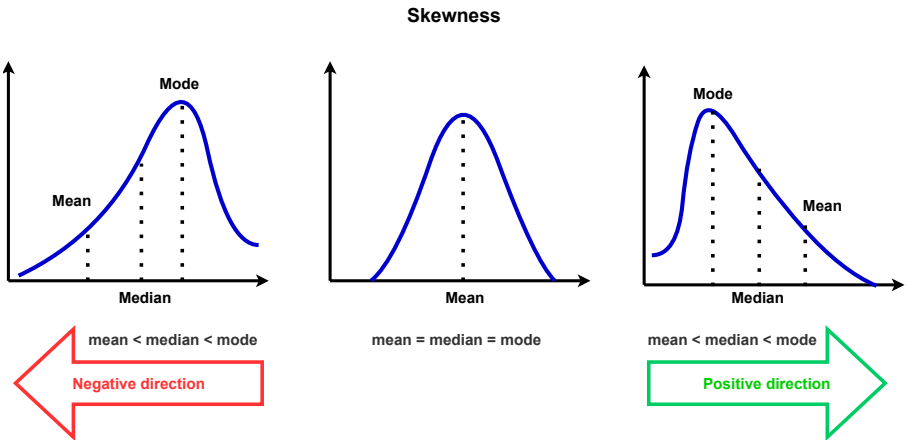

Figure 16: Two different types of Skewness value

3.2.5 Performance index:

Another approach for evaluating bearing health is proposing a performance index employing signals and characteristics of the standard operating status and behavior of the asset. This index can be efficiently used to monitor the healthy status of the component and help to find the proper thresholds in order to extract the outliers and avoid future possible failures. Finding a correlation between parameters is a way of finding a health index used and proposed by different research studies. For instance for bearing condition monitoring, it can be useful to find the correlation between factors concluded in temperature rising such as gearbox bearing and oil temperature, output power, rotor rpm, ambient temperature, hub temperature, or the correlation between the wind speed and output energy, etc, to define a performance index and employ it as an evaluation parameter to separate the healthy and faulty working conditions and consequently uncovering anomalies and forecasting the future trend. An example of Wind turbine correlation between two parameters has shown in the figures Fig.17, Fig.18.

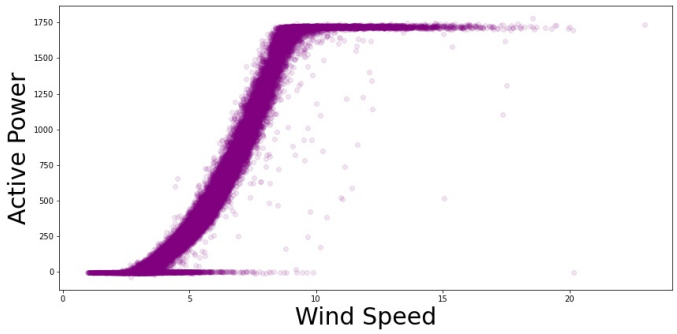

Figure 17: Correlation between output wind power and wind speed in a wind turbine based on the actual data.

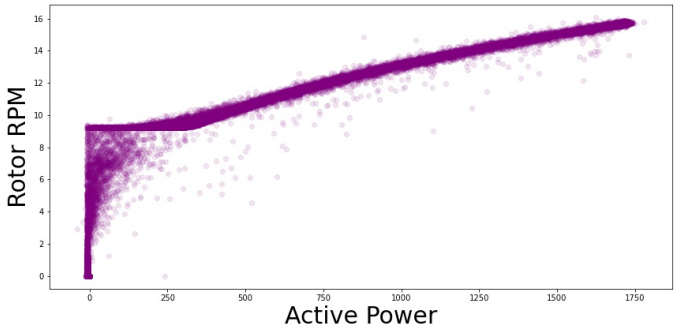

Figure 18: Correlation between active power and rotor speed in a wind turbine based on the actual data.

AS can be seen, the relation between the active power of the wind turbine and its rotor speed and wind power which is almost a constant correlation in all wind turbines can be utilized to describe a standard performance index for wind turbine status.

3.3 How to predict the bearing failure

The analysis methodologies for monitoring the bearing condition have been discussed in previous sections and the process of finding abnormalities and predicting the probability of a bearing failing will be covered in the following section. The normal and trend procedure used for bearing health prediction proposed and used in various published research and study are as follows:

1. Obtaining the useful data from the asset which can be done via various methods like Supervisory control and data acquisition (SCADA), etc.
2. Cleaning and processing the acquired raw data for dealing with missing values, dividing the dataset into dependent and independent variables, splitting the dataset into training and testing sets, etc. by applying data processing methodologies to achieve an understandable format.
3. Defining a performance index based on the related parameters and factors using methods discussed in the previous section and setting a threshold to find the outliers.
4. Building a model to train the data, which is mostly implemented by machine learning methods using the healthy part of the data or the healthy index in the previous step as the training set and train the model to recognize the normal trend of the working condition, and then test the model on the rest of the dataset to check the accuracy of the built model and finally find the best model.
5. Forecasting the future trend of the data based on the built model using machine learning algorithms and methodologies.
6. Recognizing, extracting, and evaluating the outliers and irregularities in order to discover the possible failure of the working conditions.
7. Decision making and solving the problem either by humans or by employing artificial intelligence to change the potential failure condition into a healthy state.

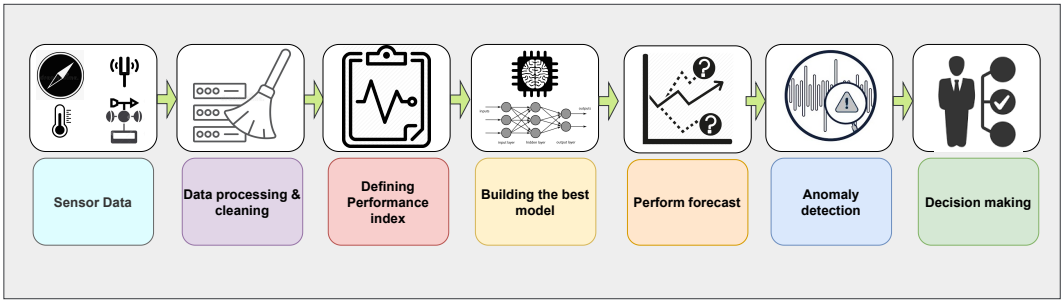

Figure 19: Failure prediction procedure using machine learning.

4 Chapter IV - Proposed Solution

In this thesis, a predictive digital twin prototype has been developed for wind farms with bi-directional communication between the physical and digital assets to receive feedback and set up commands and support real-time data visualization, data-driven variable prediction, data processing through 3D visualization, augmented reality, and cloud-based 2D GUI platforms. Along with two-way data transmission and visualization, it offers what-if scenarios simulation and analysis by adding and removing assets and changing the environment variables.

4.1 Use cases

This section explores some possible use cases in three aspects, including applications (why), data sources (what), and researchers (who).

To begin with, the research area of offshore/onshore wind farms can be treated as a software project design using top-down and bottom-up approaches. According to the analysis and surveys conducted for this project, figure 20 represents the potential use cases of the proposed platform. In a scenario where resources in the project are abundant, top-down is the best approach, however, in a scenario where time and other resources are limited, it makes more sense to use bottom-up approaches to quick accomplishment. Iterative development and maintenance are required for both methods. In this project, wake loss will be analyzed as a starting point along with a Matlab model for Wind Energy Systems. Additionally, implementing the functions on already existing datasets or models can speed up the simulation from another perspective. This project, for instance, uses an open dataset and model which will be discussed in Section 4.2.2. And lastly, applying other people's research insights and perspectives in the field of digital twin can effectively shorten the path and enhance the desired outcome.

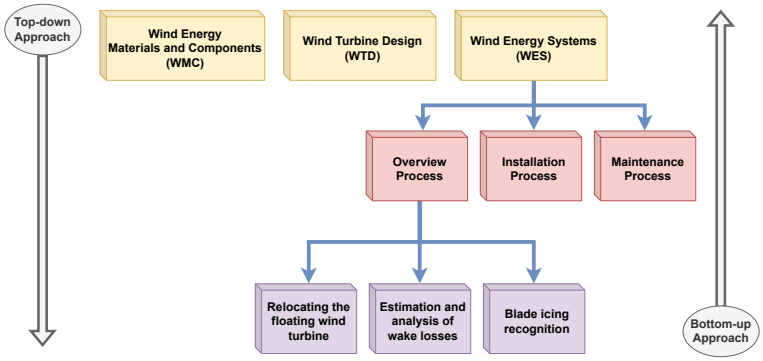

Figure 20: Potential use cases analysed for this project by considering top-down and bottom-up approach

The wind farm is designed and located on the scene to resemble the Hywind Tampen floating wind farm project with five oil rigs and eleven floating wind turbines. Hywind Tampen is an 88-megawatt floating wind power project from Equinor designed to provide power for operations at Snorre and Gullfaks offshore platforms in the Norwegian North Sea and help to reduce gas turbine energy use at the offshore oil platforms, while at the same time eliminating 200,000 tons of CO2 and 1000 tons of NOx emissions per year. This is the world's first floating wind farm powered by offshore oil and gas platforms [69].

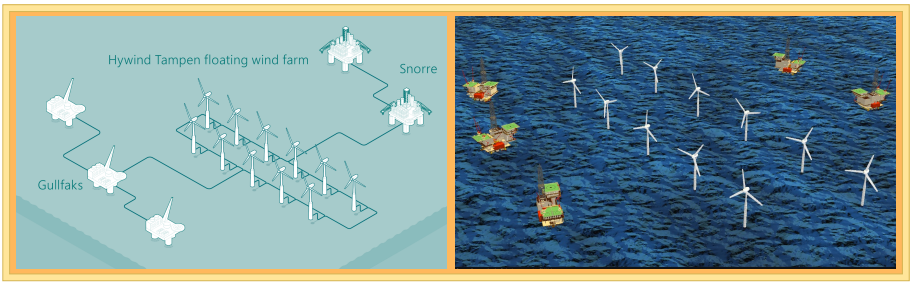

Figure 21: The left picture shows the Hywind Tampen project proposed by Equinor and the right one is the designed scene based on Hywind Tampen in the proposed project in Unity3D.

4.2 Integration

The solution prototype is an integration of simulation and visualization platforms using wind energy physic models and historical data resources to implement a seamless platform providing all the digital twin benefits to the user for evaluating the wind power energy in wind farms.

4.2.1 Renewable energy system

Through the turbine, the wind's kinetic energy is converted into rotational kinetic energy, and the generator takes this rotational kinetic energy and converts it into electrical energy, which is fed into the power grid. A turbine's swept area and wind speed are two factors that govern how much energy can be converted. Calculating the economic viability of wind farms requires knowledge of how much power and energy each turbine will produce.

Definitions of various terms can be found in the following table:

Table 1: Variables used in the model

Symbol	Parameters	Unit
P	Power	W
ρ	Density	kg/m^3
A	Swept Area	m^3
V	Wind Speed	m/s
r	Radius	m
l	Blade length	m
C_p	Power coefficient	-

In the turbine the output power can be calculated by using the follow equation Eq 7:

$$P = \frac{1}{2}\rho A V^3 \tag{7}$$

According to Professor Albert Betz, it is not possible to convert more than 59% of wind energy into mechanical energy. As a result, the Betz limit states that an ideal wind turbine can only convert 59% of wind energy into electricity [70]. This is named the "power coefficient" and is defined as:

$$C_{Pmax} = 0.59 \tag{8}$$

Figure 22 represents the max power efficiency produced by different types of wind turbines considering the tip speed ratio which is used by wind turbine designers to correctly match and optimize a blade set to a particular generator. It is measured as the following equation Eq 9:

$$Tip\ speed\ ratio\ \lambda = \frac{\text{Speed of rotor tip}}{\text{Wind speed}} \tag{9}$$

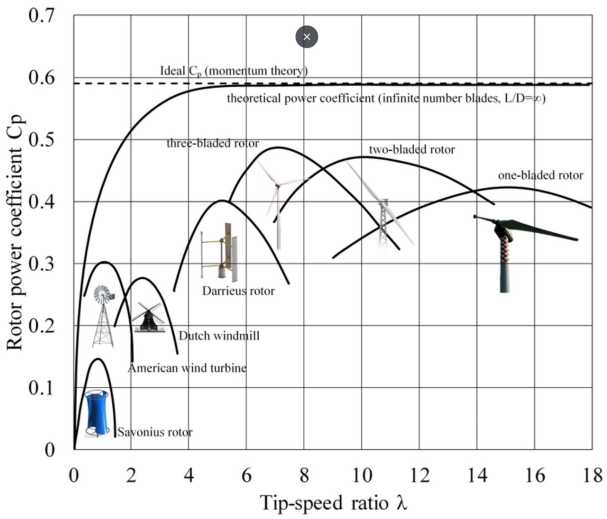

Figure 22: Betz limit for different types of wind turbines

Wind turbines, on the other hand, are unable to function at this maximum. The Cp value is specific to each turbine type and is based on the wind speed (Figure 22). Even with the best-designed wind turbines, the real-world limit is significantly below the Betz Limit, with values of 0.35-0.45 frequent.

By considering this factor the equations is given by Eq 10:

$$P = \frac{1}{2}\rho A V^3 C_P \tag{10}$$

The rotational speed of wind turbines, can be calculate from the Tip Speed Ratio (TSR) factor in Eq 9 as follow equation Eq 11:

$$Tip\ speed\ ratio\ \lambda = \frac{\omega R}{V} \tag{11}$$

where ω is the rotational speed of the wind turbine rotor (radians/second), and R is the rotor radius (metres).

Table 2: Variables used for calculating TSR

Symbol	Parameters	Unit
V	Wind Speed	m/s
ω	Angular velocity	rad/sec
r	rotor radius	m
f	frequency of rotation	Hz

Away from getting into detail, physics and study have determined that the approximate ideal TSRs for a particular blade rotor can be represented as the following table:

Table 3: Estimated TSR

TSR	Number of Blades
6-7	2
5-6	3
2-3	5

In this project, wind turbines with three blades have considered which almost have the highest power efficiency and according to the table 3 the approximate TSR will be 6 and consequently, the rotational speed of the wind turbine is given by equation Eq 12:

$$Rotational\ speed\ (rpm) = \frac{TSR \times WindSpeed \times 60}{2\pi r} \tag{12}$$

4.2.2 Data resources

The project has used a variety of data sources, which provides more options and capabilities depending on the application and requirement which are described as follows and shown

in the figure Fig.23:

- Static data: which are the data created by the user in the 3D platform to conduct various experiments and what-if scenarios that are customized for improved results.
- Live data: The real-time data received from the physical asset through the sensors like wind speed, direction, temperature, etc, used for simulating completely realistic scenarios.
- Historical data: which are used for simulating semi-realistic scenarios to conduct prediction and processing procedures. The datasets are related to actual wind turbines working over the world using the SCADA system utilized to monitor the real-time condition and information of different parts of the turbines. [71], [72] Additionally, NASA data sets have been donated by a number of universities, companies, and agencies that specialize exclusively in prognostic datasets [73].
- Simulated data: that is created with Matlab and imported with the FMI plug-in into the system from more complex physical models.

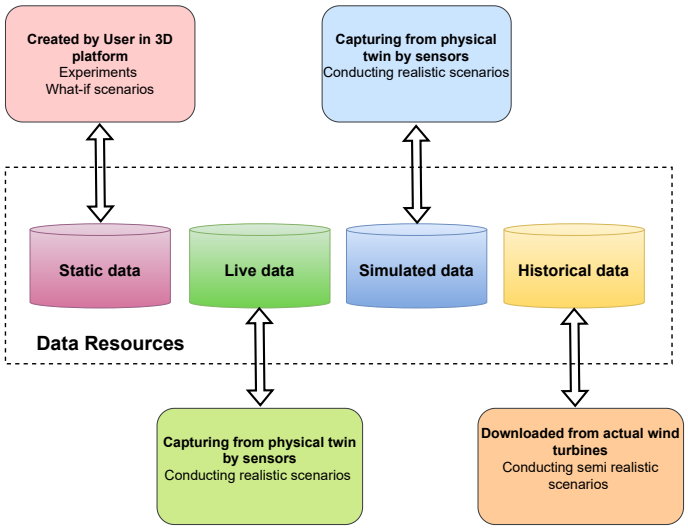

Figure 23: Data resources used in proposed solution. The figure shows the data resources and their use cases for conducting diverse scenarios.

4.3 Implementation

The implementation of the proposed predictive DT platform is divided into visualization, simulation, communication, and prediction sections. The visualization and simulation partially has been implemented in previous project in specialization course and

4.3.1 Visualization

The visualization part of the proposed project implemented in diverse platforms of 2D GUI cloud, 3D visualization, and Augmented Reality which will be discussed in more detail.

4.3.1.1 3D visualization

The 3D visualization has been implemented in Unity 3D, an interactable open-source platform that allows users to drag-and-drop assets from an inventory to the scene and set up various scenarios by changing internal and external factors. Except for the wind turbine and oil rig, which are free 3D models taken from the internet, the setting was entirely developed from scratch. A realistic ocean has been created using water textures and shaders to simulate waves, foam, and movement in response to wind speed and direction, which can be enhanced in the future by adding hydrodynamic models and more dynamic physics. The scene contains wind turbines and oil rigs to simulate the use case Hywind Tampen floating wind farm project by Equinor [69] which is designed to feed the oil platforms using wind farm output power as discussed in the use cases section before. Wind turbines are working based on the physics models mentioned in renewable energy system section 4.2.1, equation Eq.10.

To make a more precise calculation model of output power, other coefficients such as mechanical loss, electrical loss, time out of order, and wake loss are also taken into account and represented as percentage factors in the proposed model which is defined as follows equation Eq.13:

$$\mu = (1 - (\frac{MechLoss}{100})) \times (1 - (\frac{ElecLoss}{100})) \times (1 - (\frac{TimeOut}{100})) \times (1 - (\frac{WakeLoss}{100})) \quad (13)$$

Which implies the definitive model equation as following equation Eq.14:

Eq 14:

$$P = \frac{1}{2}\rho A V^3 C_P \mu \quad (14)$$

User Interface components Users are able to access the user interface in two different modes: operator mode and editor mode, based on their permission levels. Operators can control and set the wind farm and wind condition by using the available panels, sliders, input fields, and buttons as shown in Figure Fig.24.

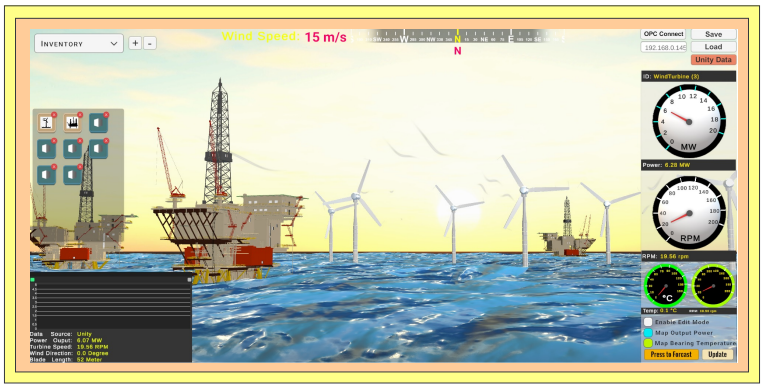

Figure 24: Controlling and visualization of desired scenario in Unity 3D platform including asset inventory and UI indicators.

The features used to implement the user interface are the widgets utilized for the dashboard such as sliders, input fields, buttons, toggles, charts, etc, and 3D models used in the

inventory panel are wind turbine and oil rig.

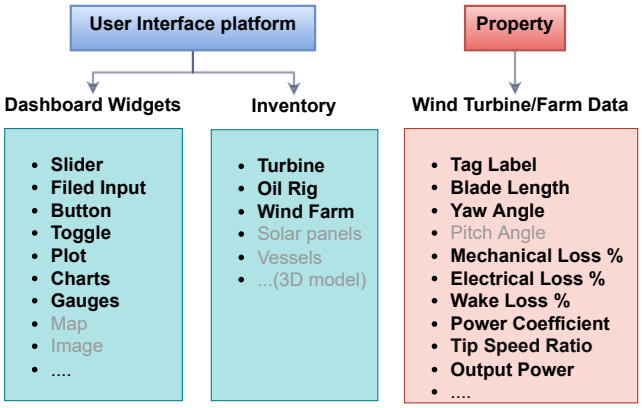

Figure 25: The components used to implement the User Interface and configuration settings.

As a result, considering the coefficients will influence the output wind turbine's power, enabling the user to adjust loss factors, as well as important variables like blade length, thus altering produced power based on the desired scenario and visualizing it on the user interface as can be seen in the figure Fig.26.

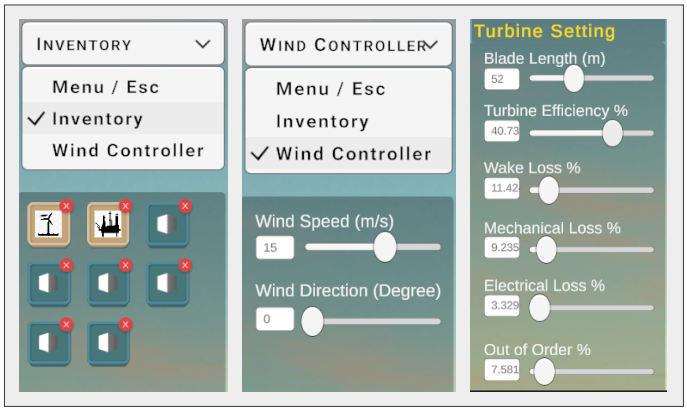

Figure 26: User ability panels. The left panel is inventory which is used for adding a new asset to the scene, the middle panel is the wind controller used for setting the wind conditions, and the right panel is turbine modification setting which is used to change the turbine's parameters

The user has the ability to switch instantly among four separate data modes explained in data resources in section 4.2.2.

The first mode is Unity3D which is the static data that are user-defined parameters in Unity3D that can be utilized and adjusted directly from the user interface and editor to determine the desired output and situation.

The second mode is FMU which is the simulated data imported from Matlab Simulink or other simulation applications using FMI plugin that allows the user to conduct more complicated experiments based on the complex imported models and consequently visualize in the Unity3D platform.

The third and most practical technique is OPCUA mode which provides a two-way communication data transferring from the physical asset to the digital asset, allowing the user to conduct diverse experiments and what-if scenarios based on the real-time data and also give the command to the physical asset simultaneously that is presenting the main concept of the digital twin.

The last mode is the actual data related to actual wind farms' historical data imported from CSV files to operate semi-realistic scenarios and investigations.

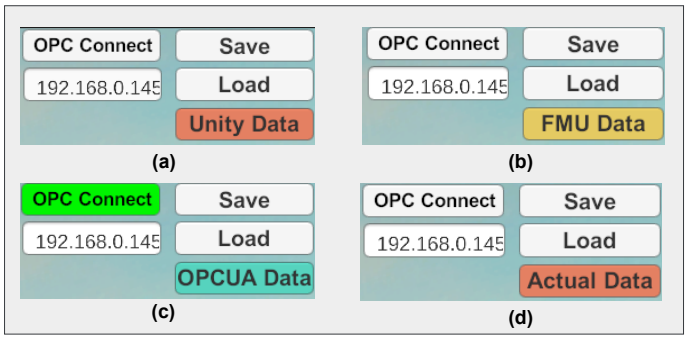

Figure 27: Data resource switch buttons are shown in the figure, (a)Unity data (b)FMU data (c)OPC UA (d)Actual data

The editor mode provides more configurations and abilities to change the parameters which can give a higher level of access to the user based on the hierarchy. The available editor mode can be seen in the figure.28 is the same Unity3D platform by adding sliders, field inputs, and other features to configure the desired scenarios.

Figure 28: The editor mode of user interface using to access more configuration settings based on the user access.

To represent the output power of each wind turbine, a thermal bar indicator on the turbine has been designed to show the current power generation in a green to red color gradient showing the minimum to maximum power. The bearing vibration and temperature also are shown on small panels on top of each turbine, showing the current value and the minimum and maximum range of the values changes which can be seen in the figure Fig.29

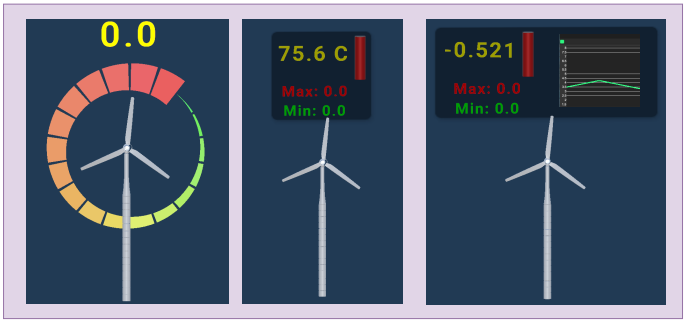

Figure 29: The left figure shows the power output indicator, the middle one shows the bearing temperature indicator, and the right one shows the bearing vibration indicator.

In the 3D visualization, the condition indicator panel located on the top of each wind turbine includes the temperature or vibration values, a small bar cylinder shows the amount of the value by its height also the color changing from green to red represents the minimum to maximum value respectively. If the temperature or vibration value exceeds the minimax values, therefore, the turbine blades color turns to red in case of passing the maximum value and an alarm sign will appear on top of the turbine, or blue if it goes beneath the minimum value as can be seen in the figure.30. The implementation of artificial intelligence can optimize the working conditions to prevent future failures. In this case, if the alarm sounds five times or stays on for 10 seconds, the turbine's rotor speed slows down and stops until the alarm turns off and the turbine returns to normal. There are of course more complicated algorithms and AI methods that can be used to optimize the abnormal or potential failure status.

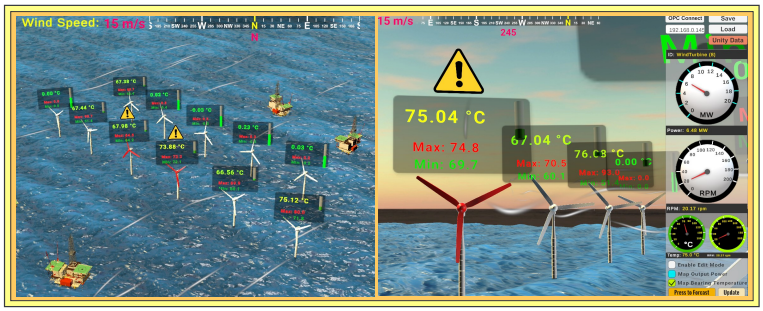

Figure 30: Condition monitoring of the turbine. If the bearing temperature or vibration exceed the healthy boundary, the turbine color turns to red with a warning sign on top of it.

Save & load scene The platform provides users with a primary save/load feature based on serialization. This allows users to save and load present projects with their exact objects, configurations, values, data, and information.

The serializable class BinarSave is created by adding name, tag, position, rotation, scale, material (texture), etc information with List type because the length of the data could vary. As a result of the complex architecture of model objects, some bugs were fixed when searching for children with the same name. The later object will only be visible in the scene if the object name is not unique when loading the scene.

4.3.1.2 2D Visualization

The 2D interactive GUI has been designed in Node-RED cloud platform which shows the live data simultaneously with other visualization platforms. It brings the ability to change the attributes of physical and digital assets through the other mobile devices and computers which makes it easy to access by the other users. The figure Fig.31 represents the implemented dashboard for this project containing charts, gauges, input fields, and indicators.

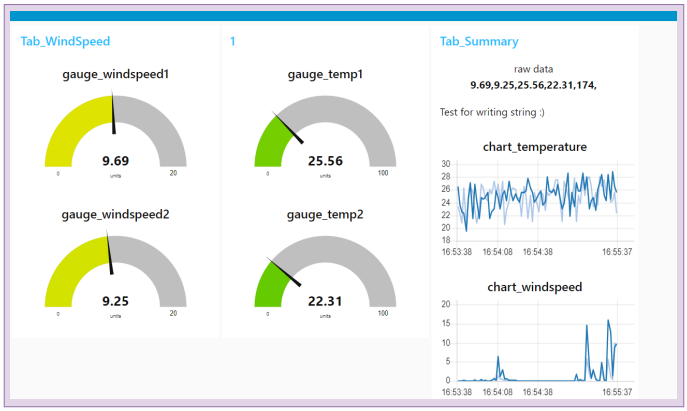

Figure 31: Designed 2D interactive GUI dashboard in Node-RED

4.3.1.3 Augmented Reality

The proposed solution is equipped with Augmented Reality in order to improve and facilitate user interaction and abilities. The concept of Augmented Reality involves blending the digital world with the physical world. This is accomplished using technology to visualize and combine them together, providing a platform where users can interact more easily. This option enables users to directly access their digital assets via smartphones for obtaining useful data or adjusting physical assets using technologies such as IoT without requiring special hardware or tools. PTC Vuforia plugin for Unity3D has been used to bring AR technology to this project. To activate AR, users need to hold their smartphone or tablet camera and focus on the image target, which can be a graphic, QR code, or 3D object, and the 3D model pops up on the mobile devices and provide the AR interaction facilities. The AR platform works simultaneously with the other visualization platforms and all the features and modifications are available to get changed and interact by the user. The implemented AR can be seen in the figure Fig.32, 33 and other related videos and images can be found in the project presentation repository [74].

Figure 32: Using Augmented Reality platform through an Android tablet.

Figure 33: Augmented reality visualization view of digital twin of wind farm.

4.3.2 Communication

As mentioned before, the main concept of digital twin is defined a platform containing a two-way communication between physical and digital parts. This project benefits from OPC UA protocol and Node-RED platform to provide such a seamless communication.

4.3.2.1 OPC UA communication protocol

OPC stands for Open Platform Communications and UA stands for Unified Architecture which is a key component of Industry 4.0 concept and the Internet of Things. Using

OPC UA, devices, and further cyber security systems in the industrial environment can be accessed in a common way and data can be exchanged across them in a similar manner, regardless of the manufacturer [75].

In this project an authenticated communication has used to provide connection between server and clients. The OPC UA servers can be created through EAExpert application or other platforms offering this service and all the clients can be connected to the available servers from other devices.

4.3.2.2 FMU/FMI

In engineering design, the use of model-based simulation and visualization is common. The Functional Mock-up Interface (FMI) is the most popular way for over 100 tools to communicate data across dynamic models, therefore it is considered as the tool communicator for this project. Model exchange (ME) and co-simulation (CS) are included in this independent tool standard. It creates a single file that contains a container and a collection of XML files, model files, and other data. Functional Mock-up Unit (FMU) is the model file. The difference between FMU-ME and FMU-CS is if the simulation tool contains a solver. [76]

In 2010, FMI Version 1.0 was released covering all of the fundamental FMU principles, such as Model exchange and co-simulation. In 2014, FMI Version 2.0 was launched and added various features, such as support for directional derivatives, and explained some of the first version standard's ambiguities.

FMI Version 3.0 beta.2 was re-released earlier this year, and it includes support for two types of clock-based simulations: Synchronous Clocked Simulation and Scheduled Execution. It was created for real-time co-simulation [77].

FMU 2.0 CS was used in this project because the original model file for the wind turbine was in Matlab format and only supported FMU CS.

4.3.2.3 Node-RED platform

Node-RED is an open-source API platform developed by IBM's Emerging Technology Services team providing a broad range of online services for connecting physical assets, including hardware, to digital assets. Using Node-RED, serial port data can be synchronized with OPC UA Server namespaces. Sensor data is connected to the digital platform by using OPC UA and serial communication blocks from Node-RED. The figure Fig.35 depicts the project communication architecture built using OPC UA and Node-RED and Arduino.

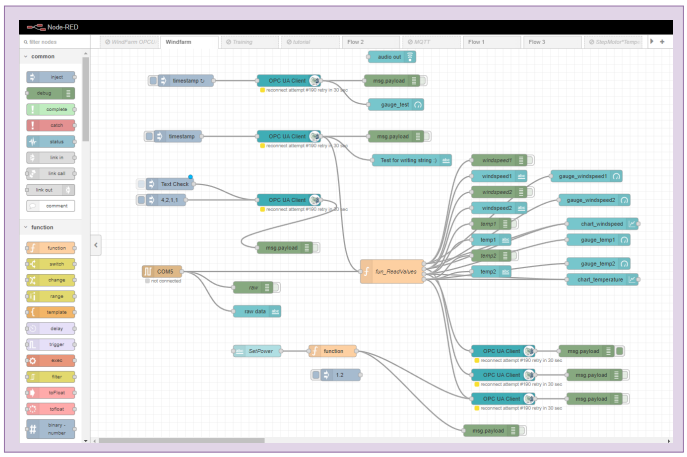

Figure 34: Node-red interface and blocks used for communication between physical twin and digital twin in the proposed project.

4.3.2.4 Arduino board

Local sensors are connected to the digital platform via Arduino UNO WiFi Rev.2 board which is an IoT hardware used for creating sensor networks. It transmits the sensor data through serial communication which is accessible through cloud platforms and WiFi devices. In this project, all sensors are connected to Arduino board and it sends the data to the PC via serial communication port and consequently, can be transferred through Node-RED platform by adding a serial port block.

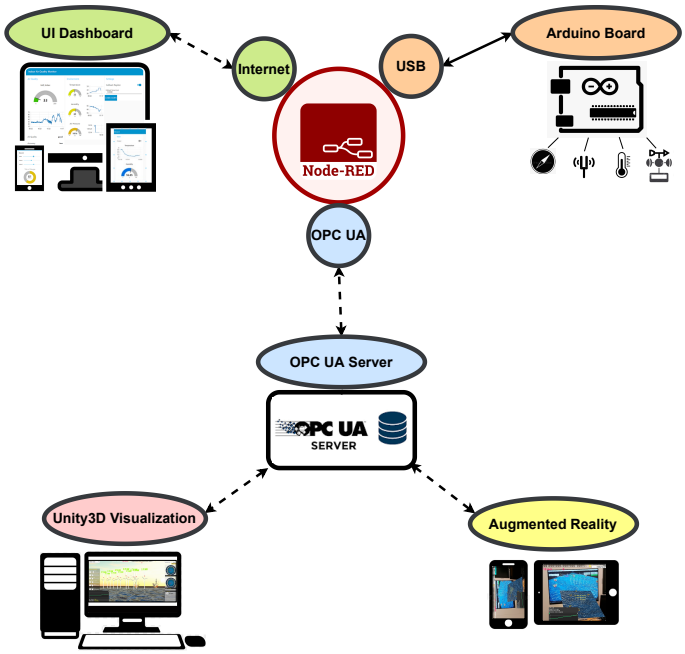

Figure 35: Communication architecture used for connecting the components applying OPC UA, Node-RED and Arduino board

The Node-RED platform interface implemented for this project has been shown in the figure.34.

4.3.3 Simulation

The simulation section serves as a link between the data sources and the visualization sections. It had the ability to turn genuine raw data into internal data variables that could be linked to any scene object. It could administer the data service and data bank in the meanwhile. All the data will be processed and calculated in Unity3D by the simulation functions which receive the data from different sources and gets the customized parameters to return the desired output and visualize them in the scene. It works by defining functions based on the wind energy physics and equations, also additional coefficients change by the user in order to simulate various scenarios. The parameters set by the user and data received from the data source get merged in Unity3D functions and turn into the output power, rotor speed, yaw angle, and blade length to represent the wind farm condition based on the designed experiment, also real-time data receiving from the physical asset.

For simulating the static data, the proposed platform uses the data created in Unity3D to calculate and returns the output which is the first way of simulation based on the data defined by the user.

In order to import and process the simulated data, Unity3D is receiving data from other simulation applications like Matlab and Simulink. The model can be created in Matlab or other applications which are supporting FMI standards, in order to make a co-simulation and transfer the simulated parameters to FMU format. The FMU file includes all the selected information from the simulation application and can be imported into Unity3D using FMI plugin which lets to read the data and use them in Unity 3D simulation functions to get the output power and other parameters.

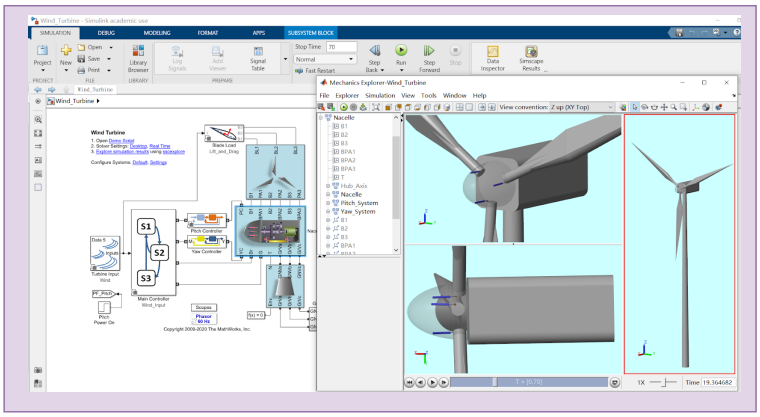

Figure 36: The complicated wind turbine model simulated in Matlab Simulink and the output data and variables can be used in Unity3D using FMI standard and plugin

Accessing the live data from the physical asset is done by implying OPC UA protocol and Node-RED platform. The real-time data captured by local sensors are sent via Node-RED OPC UA API to the Unity3D platform and they get processed and calculated in Unity3D simulation functions to return the live output depending on the receiving values.

Finally, to simulate the historical data, CSV files are imported into Unity3D and the usable data is extracted and fed back into the simulation functions, which result in a measurement of the output variables.

4.3.4 Prediction

As previously stated, the prediction algorithms chosen and implemented in this project include LSTM, SVM, Arima, Auto-Encoder Neural Network, LightGBM, and Prophet library, as explained in section 2.1.1. To choose the best algorithm, all of the approaches stated above were built particularly for the temperature and vibration problem in this research study, and their accuracy and other variables were compared to those discussed in the result and discussion chapter. Prophet library was chosen to implement and apply to the requested project once all of the trials and assessments were completed.

The procedure of implementing the prediction model of this project can be explained as follow steps:

4.3.4.1 Preprocessing and data acquisition

After importing the raw data from different resources, to train the machine learning model based on the healthy data, it is needed to clean and filter the data. The temperature data used to find the correlation and prediction comes from a real data set related to wind farms located in Gansu Province in China [78]. In this wind farm, a SCADA system measured 21 parameters at 10-minute intervals. The power rate of each wind turbine is 1800KW. The key parameters considered in this study are active power, ambient temperature, bearing shaft temperature, gearbox bearing temperature, gearbox oil temperature, generator RPM, generator winding temperature, hub temperature, main box temperature, rotor RPM, and wind speed.

The ML model will be trained using algorithms to build a relation between the inputs and outputs. Consequently, the quality of the data needs to be tested to ensure that the model accurately represents the system condition. Anomalies in data must be removed from the model to prevent the model from interpreting system performance incorrectly. Data cleaning operations are required in order to build a model representing the health state of the wind turbine. The WT system performance is based on sensors that collect data from the SCADA system; accordingly, there could be spikes in data or no data if the sensors malfunction. These errors can occur if sensors are not calibrated or if they degrade over time, yielding outliers in SCADA system. Criteria considered and used for primary data cleaning are as follows:

- When the data point for output power is negative or zero, while the wind speed is beyond the cut-in value.
- missing at least on input or output of sample points
- sample points with values out of the regular range
- sensor malfunctions and transition errors caused data loss while the wind turbine was on timeout.

The preprocessing procedure is summarized in as below.

$$\begin{cases} \text{Delete } x_i, & for\ x_i \in \text{timeout} \\ x_i = x_{i+1}\ or\ x_{i-1}, & for\ x_i \in \text{loss data} \\ x_i = 0, & for\ x_i < 0\ or\ x_i \text{ is null} \\ x_i, & for \text{ others} \end{cases} \tag{15}$$

4.3.4.2 Processing the model

Feature selection for bearing temperature

To build a model based on the healthy behavior of the wind turbine, it is required to find the most correlated variables from the input and output. Since there are a variety of parameters captured by SCADA system, it is difficult to know the related and important variables beforehand, thus this study has used the literature review to determine the optimal variable combinations for monitoring essential component system behavior along with conducting experiments to find the most correlated values in the actual data set of a wind farm showing in the figure Fig.37.

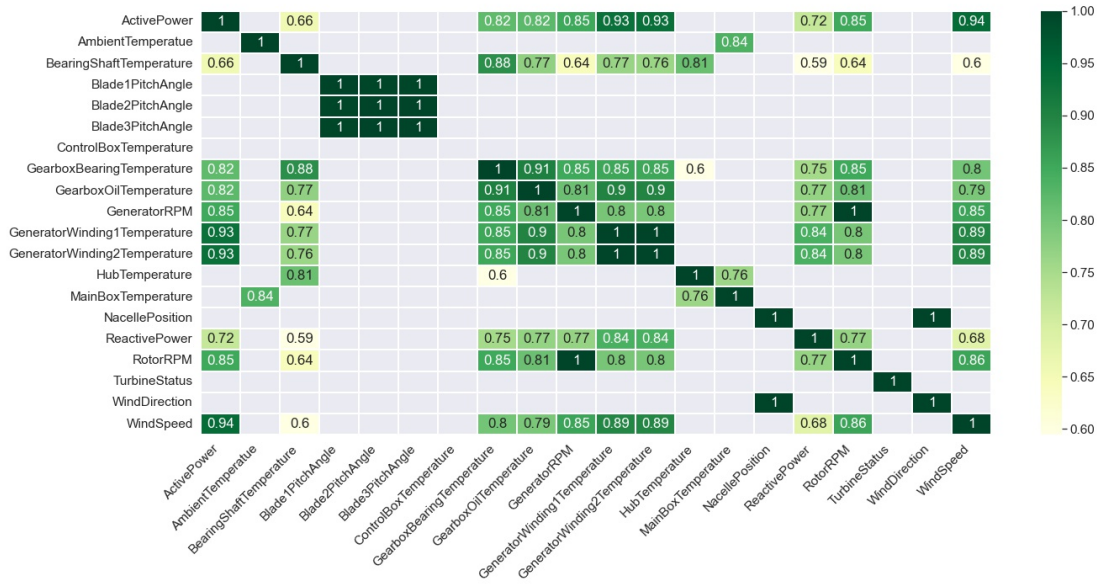

Figure 37: The correlation matrix filtered based on the correlation value that is more than 65 % for each variable against the other

According to the research studies and the result achieved from the experiments conducted on the temperature data set seen in the figure 37, the most correlated values related to bearing temperature are displayed in the table.4

These values are used to find a healthy index discussed in section 3.2.5 and train the model and the correlation diagram is shown in the figure 38

Bearing Temperature Correlation			
Generator Bearing	**Correlation**	**Gearbox Bearing**	**Correlation**
GearboxBearingTemp	0.883344	GearboxOilTemp	0.906020
HubTemperature	0.809116	GearboxOilTemp	0.906020
GearboxOilTemp	0.772887	BearingShaftTemp	0.883344
GeneratorWinding1Temp	0.765115	GeneratorWinding1Temp	0.853024
GeneratorWinding2Temp	0.763926	GeneratorWinding2Temp	0.851923
ActivePower	0.655400	RotorRPM	0.850795
GeneratorRPM	0.640947	GeneratorRPM	0.850209
RotorRPM	0.640584	ActivePower	0.818850
WindSpeed	0.596201	WindSpeed	0.800433
ReactivePower	0.594546	ReactivePower	0.75187

Table 4: The most correlated variables more than 60 % related to generator bearing and gearbox bearing temperature according to the figure 37 which is the correlation matrix implemented for the current wind turbine data set.

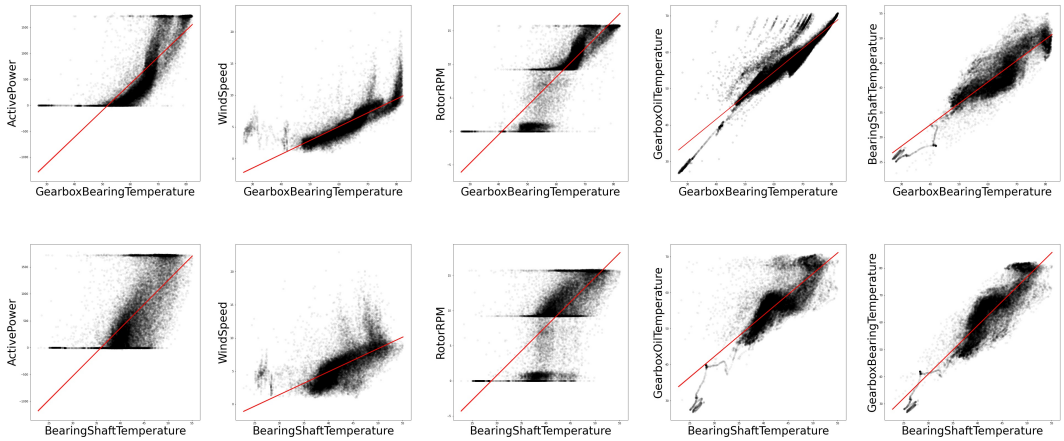

Figure 38: The correlation diagram of the gearbox bearing temperature to the most related parameters are shown in the first row of diagrams. The correlation of the generator bearing temperature to the related parameters are shown in second row of diagrams

As can be seen in the table.4 and figure.38 there are relations between different variables recorded by SCADA system that can be used to create the healthy index and train the model based on that, and consequently predict the future trend of the changes and recognize the abnormalities. This also can be implemented by considering more factors such as applying 3D correlation among three parameters of the current wind turbine which can be shown in the figure Fig.39.

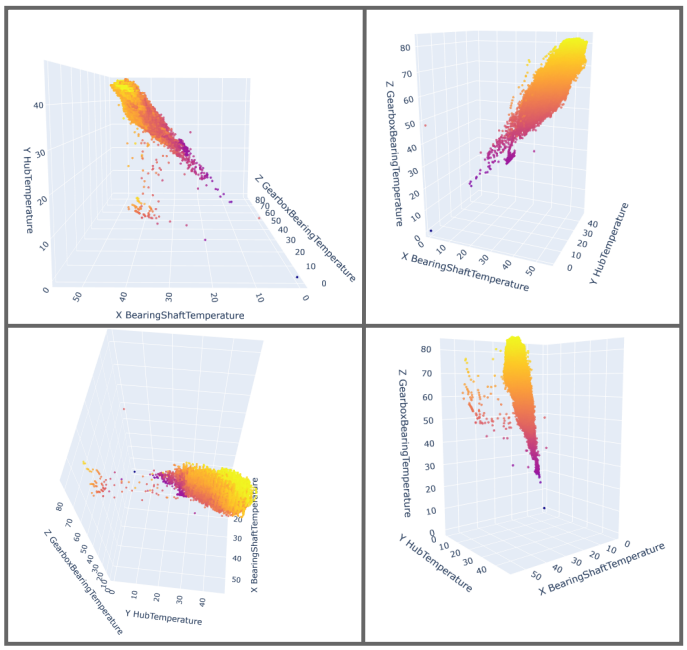

Figure 39: As can be seen there is a close relation among three parameters of hub temperature, shaft bearing and gearbox bearing temperature which also can be employed for determining a health index for wind turbine.

Defining the performance index based on the correlation of multi-dimensional variables into one-dimensional output requires conducting various experiments and separate research studies and is not the main focus of this project. Therefore, to have a performance index for temperature parameter, a simple way has been chosen which is measuring the average value of three variables of hub temperature, shaft bearing, and gearbox bearing temperature and normalizing the output and will be sent to be trained by the prediction model.

Prediction model for bearing temperature

After defining the performance index, it is time to train the model based on the healthy data and continue the prediction procedures discussed in section 3.3. The model used for this project is Prophet library which is mentioned before and the process of this stage is done as the following steps represented in the algorithm.1, This can be described as:

1-importing the historical data, 2-cleaning and preprocessing the data, 3-finding the correlation between all variables, 3-finding the most correlated factors to the bearing temperature, 4-defining the performance index based on the most correlated variables, 5-training the model on healthy data, 6-evaluating the model's accuracy by testing the trained model on the real-time data, 7-anomaly detection and check the system performance, 8-updating the index performance boundary by checking the real-time sample points if they are still moving around the healthy portion and aggregating with the historical data, 9-predicting the future trend, and recognizing the potential failures.

Algorithm 1: Bearing temperature prediction procedure

Data: Turbine dataset, using temperature correlated variables
Result: Predicting the future trend
start: import and read the csv file
foreach x_i *captured by sensors* **do**
  if $x_i \in$ *timeout* **then**
    Delete
  end
  if $x_i \in$ *loss data* **then**
    $x_i = x_{i+1}$ or x_{i-1}
  end
  if $x_i < 0$ *or* x_i *is null* **then**
    $x_i = 0$
  end
end
- find the correlation rate of variables in dataset.
- define the performance index based on the most correlated variables to bearing temperature.
foreach C_i *in* $Dataset.correlation()$ **do**
  if $corr(C_i\ , C_{BearingTemp}) > 80\%$ **then**
    $indx_i = C_i$
  end
end
$Performance\ Index = (\frac{1}{n}\sum_{i=1}^{n} indx_i).normalize$
- train the model based on performance index of the 80% of the historical healthy data.
- post-processing and test the model on the rest 20 % of the historical data.
- forecasting the future trend.
if *trend is abnormal* **then**
  - anomaly detection.
  - performing decision-making and preventive maintenance.
end
if *trend is healthy* **then**
  - Updating the performance index boundary and aggregate with the healthy historical data.
  $NewHealthyBoundary \leftarrow HistoricalData + RealTimeData$
end
end

The result will be a graph as shown in figure.40 which shows the predicted trend according to the model trained on the healthy data. As can be seen, the model has been trained based on the 80% of the healthy data presented by black dots in the figure, and the model is tested on the rest 20% of the healthy data shown with red dots. The blue line illustrates the trend of the expected values in the future based on the trained data and the light blue portion shows the acceptable variance where the variables can change.

The system employs the minimax values of the boundary to set the healthy state of the variables which shows as min and max values in the 3D visualization on top of each wind turbine. Accordingly, if the trend line (Blue line) or real-time data points move within the acceptable boundary, it means that the system is in good working condition, whereas if they exceed the boundary, it means something abnormal is happening, or if they gradually

tend to exceed the boundary, it means there may be a potential failure in the future and clearly, the values are higher or lower than the predicted values and an alarm can be sounded. If the time series is longer than two cycles, Prophet will automatically fit weekly and yearly seasonality. The add_seasonality method (in Python) can be used to add additional seasonality (monthly, quarterly, hourly, minutes, and seconds) to a sub-daily time series. Here since the target is days, the seasonality has been set to daily and as can be seen in the figure.40 The prediction trend line (blue line) represents the future forecast trend more smoothly since it has a low variance compared to changes in data points.

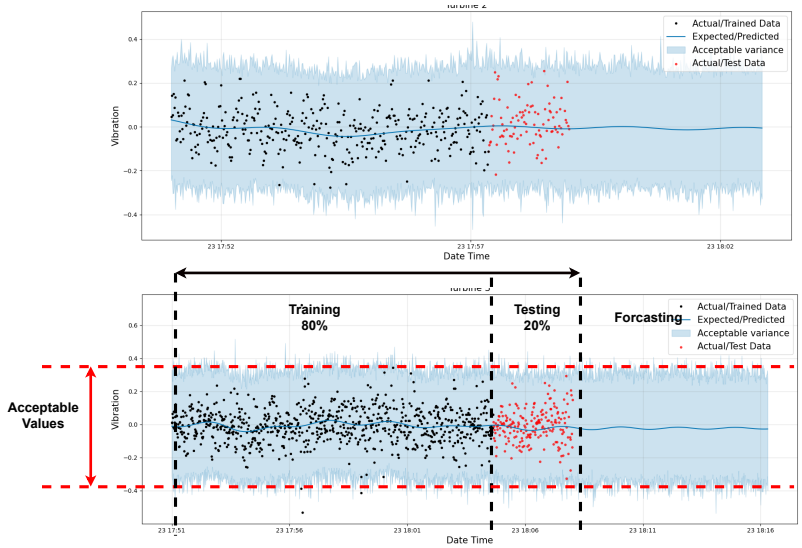

Figure 40: The black points are the actual data points of the temperature sensor. The blue line represents the expected values and the future prediction, and the light blue portion showing the acceptable variance.

Feature selection for bearing vibration

There are several methodologies for bearing condition monitoring and prediction presented in section 3.2. The described solutions have all been applied and tested to determine which is more responsive to vibrations changing and extract the abnormalities allowing the system to predict potential future failures.

The first solution is FFT (Fast Fourier Transform) discussed in section 3.2.1, which is a common and useful vibration waveform-based method utilized to monitor the bearing condition and fatigue. The FFT is a popular means of representing a signal in the frequency domain. Even with similar drivetrain designs, spectra can differ dramatically amongst wind turbines, as seen in the following case. There are several aspects that can influence the spectrum's output; nevertheless, direct comparisons may be used to establish which components contribute the most.

For simulating vibration, NASA dataset [73] was used which represents a test rig set up with four bearings attached to a shaft. Through a rub belt drive attached to the shaft, an AC motor controlled the rotation speed at 2000 RPM. With the help of a spring mechanism, a radial force of 6000 lbs is applied to the shaft and bearing. Force lubrication is applied to all bearings. The dataset consists of individual files, each containing 20,480 data points with a sample rate set at 20 kHz, each representing a 1-second snapshot of

vibration signals recorded at specific intervals. According to the result achieved after one week, it is stated that a defect in the outer race of bearing 1 was observed at the end of the failure test. The FFT analysis was performed on the mentioned dataset with almost 210000 sample points for the first and fifth days of the working period. Two different FFT results are compared and, as is clear from the figure.41, the amplitude of frequencies have changed on day 5, showing the failure of the bearings, just as it is indicated in the experiment result.

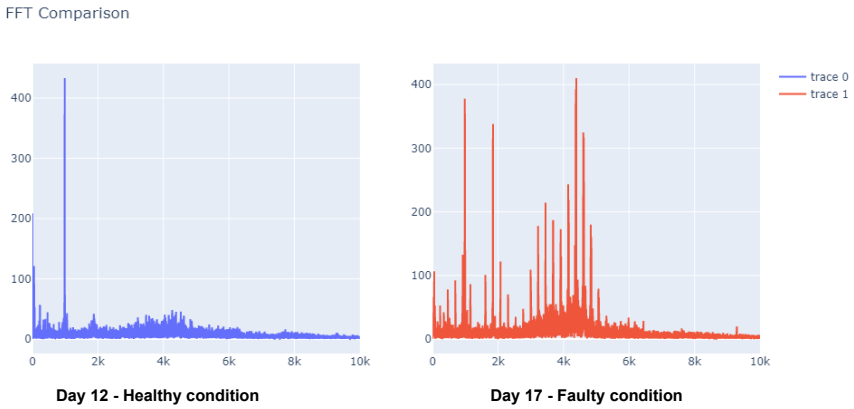

Figure 41: The left figure represents FFT result on the normal working condition on day 12 and is healthy, however, the right one illustrates the faulty data which was an inner race defect that occurred on day 5.

The process of using this solution can be described as 1-collect the data, 2-cleaning and preprocessing the data, 3-applying FFT on vibration data, 4-extracting the health data, 5-fit the model based on the healthy data, 6-assess the model accuracy on the rest of the data and compare the deviations calculated, and consequently 7-recognizing the abnormal changes in frequencies representing the BPFO, BPFI, BSF, and FTF failures discussed in section3.2.1.

The second vibration-based approach that has been employed and implemented for bearing condition monitoring and forecasting is using RMS(Root Mean Square), Kurtosis, and Skewness functions which are the vibration analysis features in the time domain. The description of each function was discussed in section 3.2.

The first feature is RMS (root mean square) value which is a suitable method since it is directly connected to the energy of the vibration wave and hence the vibration's destructive capability by taking the wave form's temporal history into consideration. It is measured by the equation Eq.4.

Additionally, Kurtosis and Skewness, discussed in section 3.2, have been applied for vibration analysis as functions used for checking if the data is normally distributed. Standard normal distributions and Gaussian distributions are probability distributions that have a mean of zero and a standard deviation of one. The normal distribution follows a bell-shaped curve and is symmetric about the mean. In relation to a normal distribution, kurtosis indicates whether the data tails behavior to check if it is heavy-tailed or light-tailed. Skewness represents the lack of symmetry. A normal distribution, or has an identical appearance on both sides of the center point. These features significantly help in the detection of outliers and the monitoring of bearing fatigue development in vibration

analysis.

The procedure of using these functions in the proposed project can be described as the following steps: 1-collecting and reading the data, 2-Cleaning and processing the data, 3-after that it needs to calculate the RMS, Kurtosis, and Skewness for sample points captured per 10 minutes, 4-extracting the healthy part of the data by evaluating the mentioned features, 5-fit the model according to the healthy data, 6-post processing and examine the model on the remain sample points, 7-forecast the future trend and as a result, identify probable bearing fatigue and abnormalities.

The algorithm.2 describes in detail how this method is implemented and applied in order to detect anomalies and possible malfunctions.

Algorithm 2: Bearing Vibration prediction procedure

Data: NASA bearing vibration dataset
Result: Predicting the anomalies and future trend
start: import and read the csv file
foreach *variable captured by sensors* **do**
  if $x_i \in$ *timeout* **then**
    Delete
  end
  if $x_i \in$ *loss data* **then**
    $x_i = x_{i+1}$ or x_{i-1}
  end
  if $x_i < 0$ *or* x_i *is null* **then**
    $x_i = 0$
  end
end
foreach *dataset captured in 10 minutes* **do**
  - calcualte RMS, Kurtosis and Skewness for N sample points captured in 10 minutes.
  $rms = \sqrt{\frac{1}{N}\sum_{i=1}^{N} X_i^2}$
  $kurt = \frac{1}{N}\sum_{i=1}^{N}(\frac{X_i - m}{\sigma})^4$
  $skew = \frac{1}{N}\sum_{i=1}^{N}(\frac{X_i - m}{\sigma})^3$
end
- extract the healthy data and define a normal trend for each calculated feature over two weeks (RMS, Kurtosis, Skewness). - fit the model based on the 80% of the historical healthy data for each feature (RMS, Kurtosis, Skewness).
-Post processing and test the model on the rest 20 % of the historical data.
- forecasting the future trend.
if *trend is abnormal* **then**
  - anomaly detection.
  - performing decision-making and preventive maintenance.
end
if *trend is healthy* **then**
  - Updating the normal trend and threshold and aggregate with the healthy historical data. $NewHealthyThreshold \leftarrow HistoricalData + RealTimeData$
end
end

The figure.42 shows the results of calculating and predicting the Kurtosis feature of bearing

vibration for 7 days of operation. In this case, the model has been trained using the healthy data displayed as black dots, and it was expected that future trends in the normal distribution would follow the blue line within the boundary, but the actual data, displayed as red dots, deviated between days 5 and 6 from the normal trend, which indicates bearing fatigue.

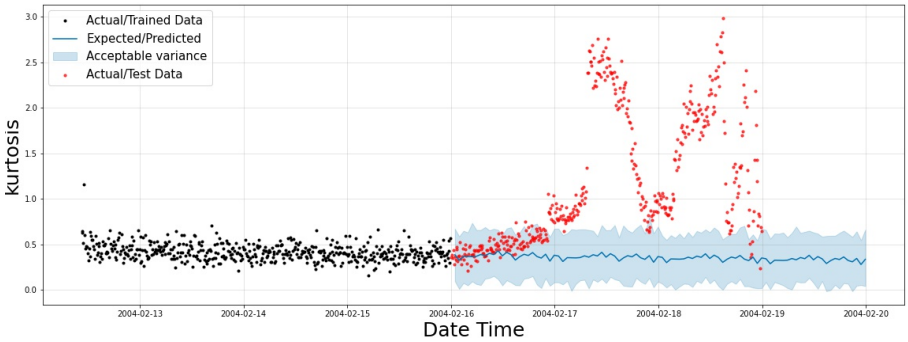

Figure 42: The graphs represent the Kurtosis of 20400 sample points captured by the vibration sensor every 10 minutes, and shows Kurtosis value of bearing vibration over two weeks.

4.3.4.3 Prediction evaluation

After the model has been trained and tested, it is time to assess its accuracy using metric coefficients of determination ($R2$), mean absolute error (MAE), and root mean square error (RMSE) (RMSE). The R-squared indicates how well the data points correspond to the fitted line. RMSE is used to calculate the average error magnitude. The MAE is defined as the average of the absolute differences between the forecast and the related observation over the verification sample [79].

They will be calculated by the following equations:

$$R^2 = \frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)/\frac{1}{m}\sum_{i=1}^{m}(y_i - \bar{y}_i) \tag{16}$$

$$RMSE = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(y_i - \hat{y}_i)^2} \tag{17}$$

$$MAE = \frac{1}{m}\sum_{i=1}^{m}|(y_i - \hat{y}_i)| \tag{18}$$

Assuming m instance numbers of data points in the test set, y_i is the measured value, $\hat{y}_i$ is the predicted value, and $\bar{y}_i$ is the mean of the calculated value.

4.4 Incorporation

To federate all of the implemented sections including simulation, visualization, and prediction, it is needed to merge all of the into one core platform. Unity3D is the main platform

of this project utilized to simulate and visualize the scenarios based on the functions defined in C# for wind turbines and wind energy to measure the output variables. The prediction part is mostly implemented and calculated in Python which supports valuable libraries for data processing, and machine learning algorithms, as well as illustrating the output charts and graphs more efficiently and intelligibly. The process of working can be described as the following steps:

4.4.1 Data simulation and transferring

Since there is no real connection for the wind farm in this project, it is required to utilize two types of historical and real-time data to simulate the realistic scenario and wind farm. The historical data collected from an actual wind farm [71], [72] are used to simulate the wind turbine bearing temperature and vibration with real data which represent the realistic signals of wind turbine operation. The working operation can be simulated in different time intervals based on the editor setting in Unity3D from milliseconds to hours. It provides an artificial illustration of a natural scenario of the wind farm seems that is working in actual time.

The real-time data represents the real-world scenarios and is used to provide the wind speed and ambient temperature data from actual local sensors, as well as to measure the performance of the implemented AI optimizer and prediction algorithms during real-time operations. The real-time data is collected from the local wind and temperature sensors connected to the Arduino board. The board is connected to the PC by the serial port to send the captured data to the system. The received data is transferred to the Unity3D and other platforms through OPC UA protocol using Node-RED as discussed in section 4.3.2. In Node-RED a serial port block has been added to receive the collected data from Arduino board, and send it to OPC UA client block which is connected to the main OPC UA server. This data can be transferred and used by the other OPC UA clients. There are two clients employed to transfer data among the available platforms. The first one is the client made in Node-RED used to receive the sensor data and also communicate to 2D GUI dashboard. The second one is the client created in C# in Unity 3D used for communicating to 3D visualization and Augmented reality platforms.

Additionally, the data can be sent from other clients, such as a command prompt from another platform, so that the Arduino can receive the command or feedback that will affect the other physical assets making it a two-way communication digital twin platform.

The communication topology of the system and how the data is transferred between the platforms are shown in figure.43.

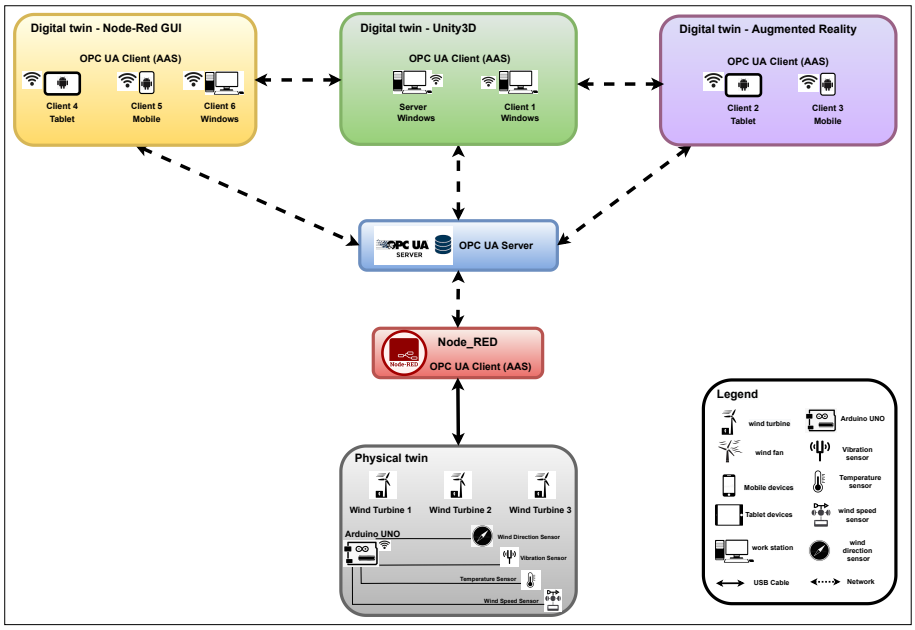

Figure 43: The figure demonstrates the communication of the components and available platforms of the system. The data is transferred through OPC UA server and clients.

4.4.2 Wind farm simulation scenarios

Simulation of the real-world wind turbine scenarios is done by the defined wind energy-related functions in the Unity3D, C# scripting. The data is received by the OPC UA client in Unity3D and it is injected into the defined wind energy functions and converted to the variables which are used for configuration and adjusting the desired scenario. To provide the natural wind farm scenario, the historical data is read from the CSV files to make the artificial presentation of the bearing temperature and vibration of the wind turbines. Each wind turbine has its own temperature and vibration changing per time interval set by the user. By starting the Unity3D, the system starts to read the bearing temperature and vibration data from CSV files and sets it for each turbine. Therefore, a realistic system will be provided to apply different what-if scenarios and prediction methods showing the wind turbine bearing condition and other desired information. Simultaneously the wind speed, wind direction, and ambient temperature are measured from the local sensors of the physical twin. All of these data are available on the wind farm which simulates the wind farm operation in the real world and the user can conduct the desired experiments on the wind farm such as changing the weather conditions to check the output power or rotor speed, the configuration of the turbine's blade length, efficiency coefficient, predict the future trend of bearing working condition, etc.

4.4.3 Forecasting the bearing condition

To predict the bearing condition of each wind turbine in the current scenario running in Unity3D, there is a forecast button on each wind turbine setting panel, which can be pressed by the user to start predicting the bearing temperature or vibration by considering the current situation and collected data. By pressing the forecast button, Unity3D triggers the Python script which runs the prediction methods for that turbine and starts to train the machine learning model based on the data collected on each turbine and represents

the output forecast in a chart, along with the healthy boundary that each turbine can work within. All the prediction calculations and output charts are presented by Python scripts.

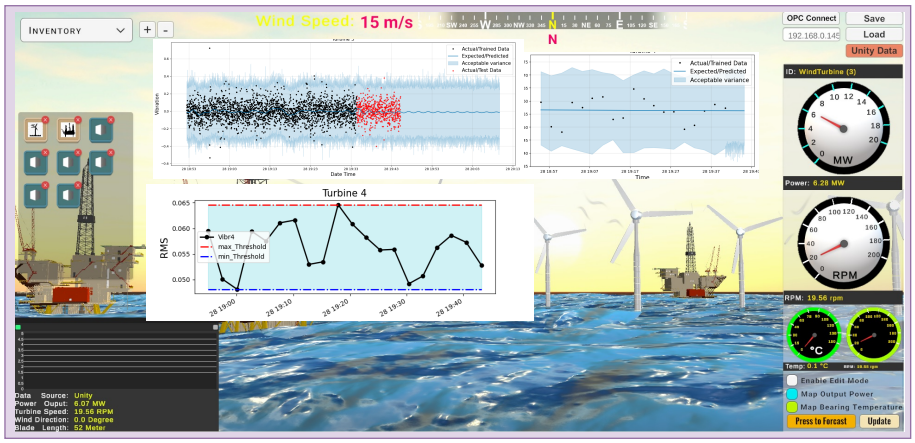

Figure 44: The system predicts the bearing condition based on the collected data from each wind turbine once the forecast button is pressed, and displays the results on the screen.

5 Chapter V - Result and Discussion

This chapters will discuss the results from the implemented digital twin, simulations, and prediction algorithms employed for forecasting the wind turbine future condition.

5.1 Simulation results

There is a hybrid-driven architecture proposed [80] that combines data-driven and model-driven architectures. To collect raw data from various sources, it will have a single database and a service management center. When the script variable is mapped to the data, it can be used to visualize the data directly (visualization data), simulate (Simulation data), and exchange the data with the model (Simulation variables). With the simulation and visualization platform, it is more flexible to link the data to any object. Figure.45 represents the model-driven and hybrid-driven architecture used for the proposed project.

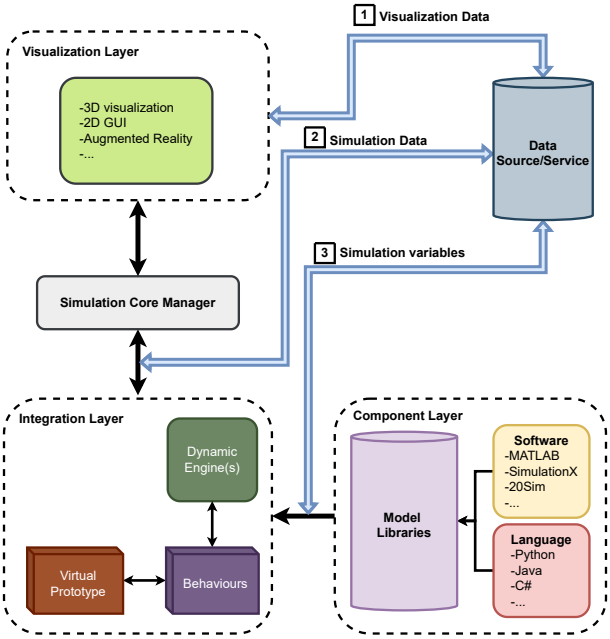

Figure 45: The Model-driven and hybrid-driven architecture, used for data management, and collaboration between simulation and visualization cores.

In this project, historical data from the wind farm is artificially stimulated to mimic natural scenarios. The simulation is completely close to the/ real-world operations providing real-time data for bearing temperature and vibration, and of course, some fatigue and failure will happen at the end of the data set which is useful to evaluate the employed algorithms and functions for prediction and condition monitoring of wind farm. The functionality of the algorithms and equations mentioned in section 4.3.2 used for the output power and rotational speed of the rotor based on the wind speed and other coefficients are giving pretty accurate results. As can be seen in the figure.46, the result calculated by the Unity3D based on the equation.14 is so close to the values calculated by the wind turbine calculator website [81] showing in the figure.47.

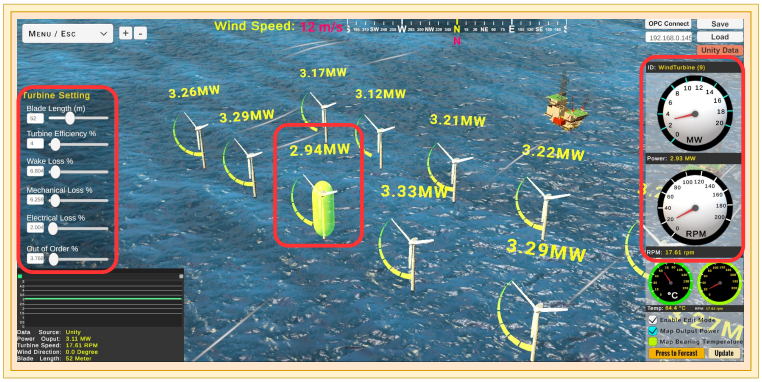

Figure 46: The simulation result of wind power and rotational speed is based on the blade length, turbine efficiency, and other coefficients in the Unity3D.

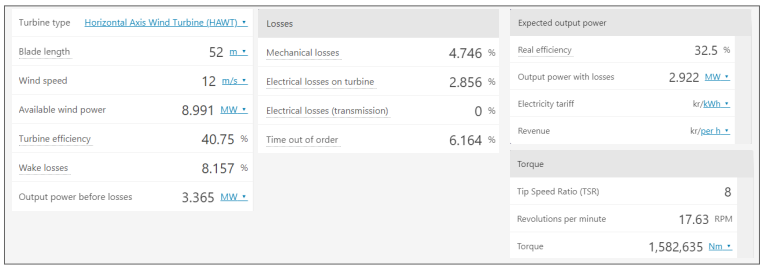

Figure 47: The calculation result of wind power and rotational speed is based on the blade length, turbine efficiency, wake loss, mechanical loss, electrical loss, and being out of order coefficients the turbine calculator website.

The only thing which makes it a bit different than the actual value is wake loss. Calculating wake loss requires aerodynamic calculations, equations, and more features which is a separate case study that should be focused on. In this project to simulate the wake effect, the distance between each turbine and the source of the wind is measured and times the output power as a coefficient. To find the best-fit equation to the actual wake effect, several experiments were conducted and finally, the following equation.19 has been chosen to simulate the real wake effect in the project. The following table.48 shows the small difference between the actual calculations and Unity3D functions of the output result because of wake loss.

$$WakeLoss = Distance(Turbine, WindZoneSource)/6; \tag{19}$$

Feature		No.	Unit	Accuracy
Blade Length		52	m	
Turbine Efficiency		40	%	
Wake Loss		8.157	%	
Mechanical Loss		4.746	%	
Electrical Loss		2.856	%	
Out of Order		6.164	%	
Output Power	Actual calculation	**2.922**	MW	**100%**
Rotor speed	Actual calculation	**17.63**	RPM	**100%**
Output Power	Unity3D Calculation	**2.93**	MW	**99.70%**
Rotor speed	Unity3D Calculation	**17.618**	RPM	**99.90%**

Figure 48: The table shows the minor difference between the actual calculation and implemented functions in Unity3D.

Therefore, all the data collected from available platforms is calculated with high accuracy which makes a semi-real scenario and can be used for conducting various experiments to achieve useful results.

5.2 Co-simulation results

There are typically two methods to connect the professional simulation tool with the architecture that has been created. For example, the simulation program Matlab provides

communication protocols such as TCP ([82]), UDP ([83]), and model file exchanging such as FMI discussed in section 4.3.2.2.

TCP and UDP modules were used to conduct communication tests, which showed that they functioned well with the platform. However, there are two significant drawbacks that force us to adopt FMU as the proposed co-simulation approach.

The first is that it is difficult to maintain the communication module operating while the Matlab module is simulating. It is possible to run it at the same time as the module simulation. Because simulation duration is usually finite, it obviously could not allow real-time visualization. The second is that simulation time synchronization is extremely difficult to support. To make sure that the FMU runs properly, a timestamp has been set for it. It's possible that the communication module will start before or after the simulation program. Finally, in this project, the communication module is ineffective as a means of co-simulation.

5.3 Prediction results

The prediction methodology implemented for the proposed project and the metric coefficient evaluation have been explained in previous section 4.3.4. Several experiments were conducted to measure the prediction accuracy and the output results. The procedure of the forecasting bearing condition is collecting the data from the physical twin local sensors, finding the related and useful factors to define the performance index and healthy boundary, and starting to train the model on the performance index based on the 80% of the collected data, testing the model on the rest of the data, evaluating the model accuracy. All of the processes and evaluations will be explained in more detail by showing and comparing the output results.

For daily intervals, the prophet library utilized for prediction is fairly reliable. The temperature of four bearings is observed on a daily basis, and then the prediction technique is used. The model's accuracy is demonstrated by the accuracy metric factors. The daily interval for predicting bearing conditions based on temperature is fairly dependable and produces good results. AS can be seen in the following figure.49, the linear regression plot, residual of regression plot represents the model trained on the daily interval average temperature data set produced a high accuracy model to predict the future.

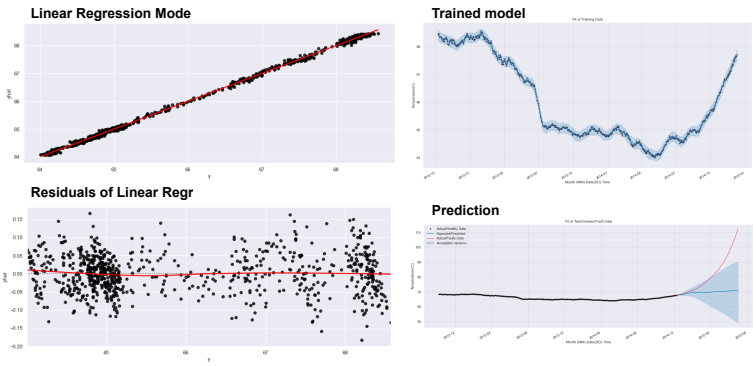

Figure 49: Left figures are linear regression model fit, and residual of linear regression of the predicted values and actual values, showing the high accuracy of the model, and the right ones are the model trained on the healthy data and the predicted trend of the values.

The model accuracy metrics of R2, RMSE, and MAE, also the correlation between the prediction and actual values shown in the figure.50, demonstrating the model's high efficiency.

Prediction and Actual Value Correlation

	yhat	y
yhat	1.000000	0.999301
y	0.999301	1.000000

Model Accuracy Metrics

R2	**0.9986**
RMSE	**0.0029**
MAE	**0.0430**

Figure 50: The table on the left shows the correlation between the actual values and the prediction values. The right one presents the model accuracy metrics, represent the high reliability and accuracy of the model and prediction values.

For vibration prediction, according to the NASA data set [73] description mentioned before, bearing No.1 will face the outer race failure at the end of the bearing operation. After aggregating the data collected for two weeks, the RMS, Kurtosis, and Skewness functions of the whole data will be calculated which represent 984 samples for each factor. Every data point is the RMS, Kurtosis, or Skewness of almost 21000 sample frequencies captured every ten minutes. The overall condition of the bearings has been checked by measuring the standard deviation of each factor, which depicts the degree of variability or dispersion of a set of values. The following figure.51 shows the standard deviation measured for each factor. The standard deviation of bearing 1 (b1) is clearly higher, which indicates the data has been distributed over a wider range and an abnormality in the data distribution, which is the outer race failure as described in the data set description.

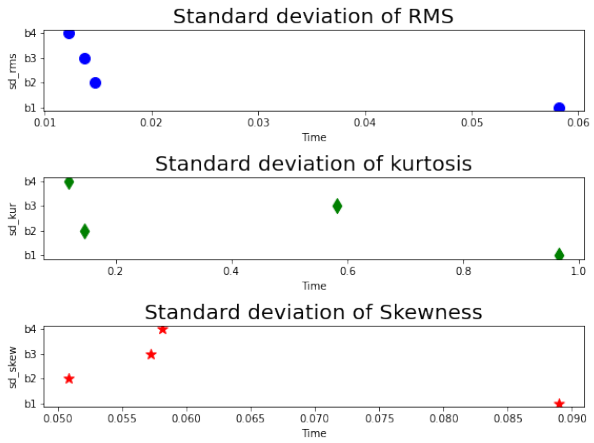

Figure 51: The standard deviations of the RMS, Skewness and Kurtosis for the four bearings that were captured over a two-week period.

In the next step the RMS, Kurtosis and Skewness values were measured for the sample points shown in the figure.52. It is evident from the charts that the normal trend of the first bearing (red line) has changed around day 5 while the rest of the bearings are working in the normal condition. It means by using the bearing analysis features, the anomalous behavior of the bearing can be detected a few days before it becomes serious.

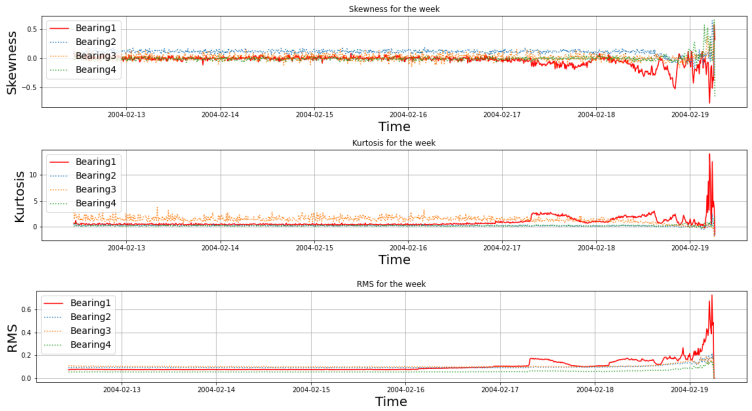

Figure 52: Each point in these graphs reflects the RMS, Kurtosis, and Skewness of 20400 data points taken by the sensor every 10 minutes, and the figure displays the RMS, Kurtosis, and Skewness value of bearing vibration over two weeks.

The data set includes healthy and faulty data. The healthy data is extracted from day 1 to day 5, and after that is considered the faulty data to test the model and check the prediction performance. The model is trained on the healthy data and the resulting output is as following figure.53.

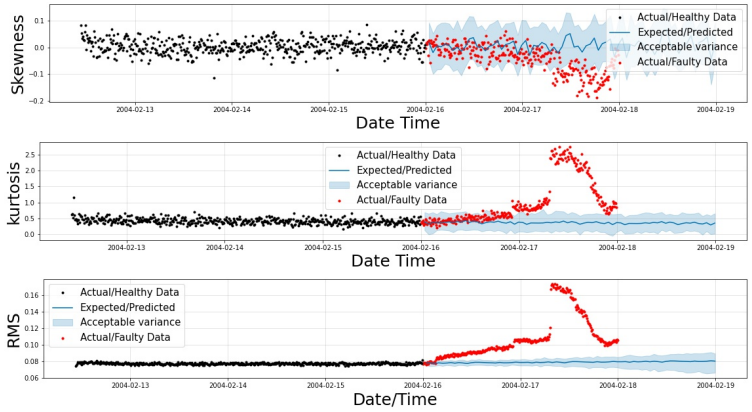

Figure 53: The prediction of RMS, Kurtosis, and Skewness of 20400 data points taken by the sensor every 10 minutes. The blue line represents the expected future trend based on the historical data and the blue portion shows the acceptable variance. The black dots are the actual historical values and the red dots are the actual values after prediction.

5.4 discussions

The accomplished results demonstrate that this approach is an effective way to reveal the abnormalities and potential bearing failure during real-time operation. However, in order to improve the system's dependability and reliability for the given prediction outcomes, the most efficient factor must be chosen as the performance index. Observing the output result of vibration predictions, it is apparent that all RMS, Kurtosis, and Skewness functions show an unnormal distribution and of course, the bearing is in the abnormal state before the serious problem for the bearing occurs. RMS function appears to be the most accurate and sensitive factor; it revealed bearing anomaly operations more than one day before the actual failure, while Kurtosis and Skewness revealed the abnormal distribution of data

after RMS, less than one day and even hours respectively. he figure.54 clearly shows the matter of fact.

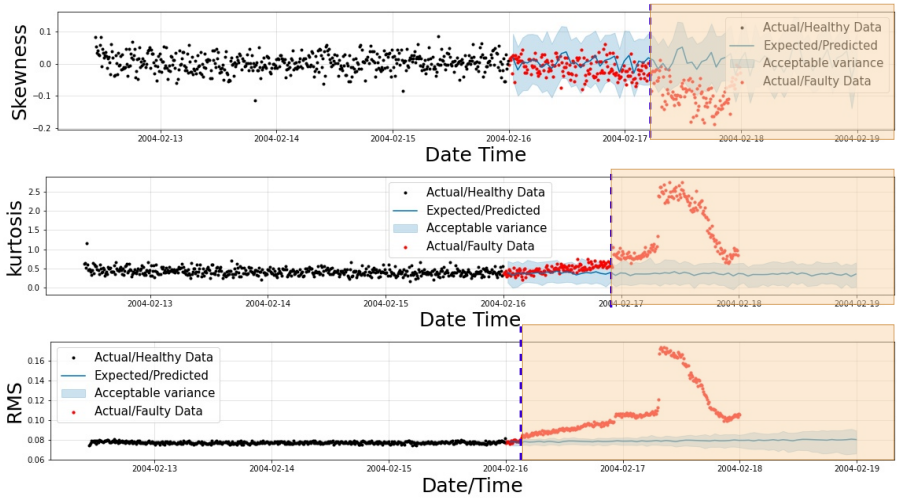

Figure 54: Comparing the applied functions to predict the bearing failure. As can be seen, the RMS is more sensitive to representing the data distribution than Skewness and Kurtosis.

Therefore, the RMS function has been selected as the main vibration performance index for the proposed project. The process is to first calculate the RMS value for a specific time interval of the historical healthy data, then find the acceptable variance and set the minimum and maximum threshold values. Afterward, the real-time data collected from the sensors are directly used by prediction algorithms to be evaluated against the healthy data and check if they are moving inside the healthy boundary, and update the historical healthy data or exceeding the boundary which indicates that something abnormal is happening and it needs to perform decision making or employing artificial intelligence algorithms.

The RMS of data was calculated by using a percentage interval which was set at 5% of the length of the whole sample. Consequently, regardless of the size of the data set, the system calculates the RMS value of the bearing vibration data at any time interval, regardless of how much data is available.

Refer to the following figure.55 which shows the applied prediction method on 6000 measurement points from the sensor, followed by the calculation of the RMS value for 5% of the total length of the data, which means 1 RMS value for every 300 sample points. As can be seen, applying the prediction directly to the captured bearing vibration data results in a normal trend without raising any suspicion of any abnormality in the data. The predicted value of the calculated RMS however shows an upward trend in the future, which indicates an increase in the RMS over time which represents that there is the potential of bearing failure in the future, and as previously stated, a bearing will fail at the end of its operation. Therefore, it is evident that the implemented algorithm based on the RMS value is so sensitive and reliable in recognizing the potential future failures.

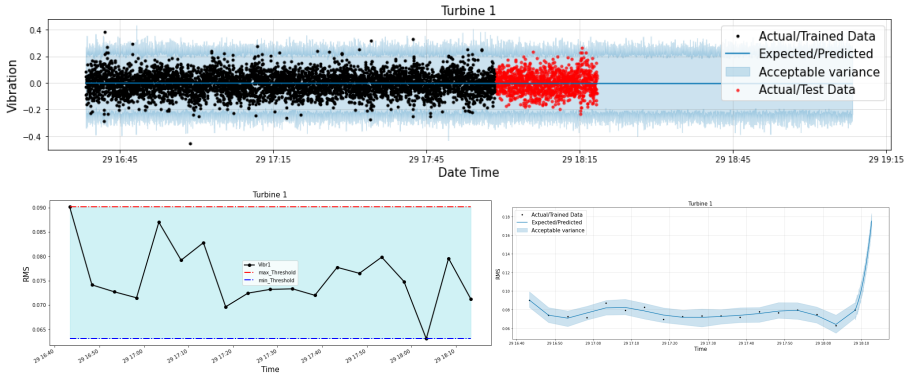

Figure 55: As seen in the top chart, the prediction algorithm is applied to all bearing vibration data points. At the bottom left, the chart displays the RMS value calculated for 5% of the total healthy counts and sets the minimax threshold. The chart on the right bottom illustrates how a predictive algorithm was applied to the RMS values to determine the future trend.

For temperature prediction, the performance index was selected based on the values that are most correlated, which are gearbox bearing temperature, bearing shaft temperature, and hub temperature. Basically, the system computes the average of the correlated values per day, clears the outliers, normalizes the result, and assigns that value to represent a healthy index. The model is now ready to be trained on the healthy part of the calculated index, using the prediction algorithm to determine the trends and anomalies of the future. It is beneficial to take the daily average temperature since the method works better for a daily interval and produces more useful results than conducting the forecast in smaller time intervals which can be seen in figure.49. It is necessary to establish a very precise performance index in order to anticipate the temperature based on the performance index in hourly or minute intervals. Because more experiments need to be conducted and research studies are required to come up with a temperature correlation performance index, which is a separate issue, yet can still be used to complete the proposed prediction goals. The applied prediction to the performance index chosen for temperature in seconds interval is shown in the figure.56. As can be seen, the future trend (blue line) is upward, and the actual values are likewise increasing towards the end of the simulation, indicating that the forecast was correct and that the probable failure was predicted before it occurred.

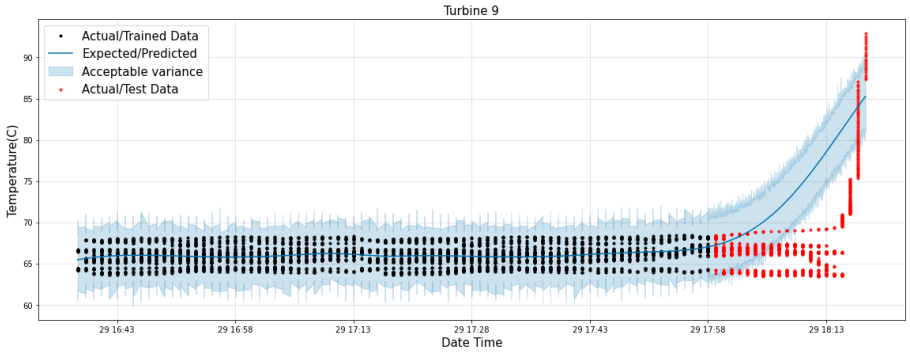

Figure 56: The chart represents the temperature prediction. The blue line depicts the anticipated future trend based on past data, while the blue portion depicts the acceptable variation. The black dots represent the actual historical values, while the red dots represent the predicted values.

The major purpose of this project was to create a digital twin prototype that could be used to make real-time predictions to recognize potential failures, and schedule adequate maintenance to mitigate risks and costs associated with downtime. The implemented prediction techniques are based on the best available solutions and methodology, and they work well in the proposed prediction digital twin platform. However, all of the prediction methods can be improved in the future by including more robust solutions.

6 Chapter VI - Conclusion and Recommendations

Wind power is a form of renewable energy. As well as being non-polluting and inexhaustible, it can reduce the use of fossil fuels that contribute to global warming by reducing greenhouse gas emissions. A bonus is that wind energy is local energy since it is available nearly everywhere in the plant, which reduces the need to import energy and creates wealth and jobs in the region. This is why wind energy is conducive to sustainable development since it can be produced efficiently and used.

With the advent of digital twins in the wind energy sector, wind turbines can now be efficiently developed, maintained, and improved to ensure safety, reliability, and peak performance while also helping to make better decisions regarding their design.

A very important function of the industrial sector is keeping track of the asset condition, as unexpected failure can have very expensive and risky consequences, including loss of production, costly replacement of component parts, or even a major environmental crisis.

Rotating machines are entirely dependent on bearings, which are almost always the primary cause of equipment failures. Therefore, bearing condition has become a significant research focus in the last few years.

When it comes to bearing monitoring, predictive maintenance procedures are employed to identify probable faults and failures. This allows maintenance to be scheduled ahead of time. Vibration and temperature analysis can be especially helpful in detecting bearing deterioration.

In this project, an intuitive prototype of a predictive digital twin of the wind farm was proposed providing two-way communication between physical and digital assets through different platforms. A wind farm analytics platform allows users to collect, visualize and analyze data in real-time to improve predictive maintenance strategies, enable better decision making, reduce potential failures, and improve reliability. The proposed platform is an integration of visualization, simulation, and prediction methods and applications unified in the Unity3D platform.

The process commences by collecting real-time data using physical sensors, simulation results, and historical data to create a realistic wind farm scenario for investigation and exploration. Diverse capabilities are offered through 2D,3D visualization, cloud-based, and augmented reality platforms which can be employed depending on the desired objectives and requirements. The proposed and implemented machine learning prediction method yields decent results by employing performance indices obtained from experiments and research studies.

The proposed framework can be enhanced in terms of efficiency and robustness by adding and implementing calculations for the other components of the wind turbine. Improving the forecast results by updating the prediction algorithms and finding a good performance

index based on the correlations of the variables. There is the capability of fine-tuning different parameters in the framework to get the optimal parameters for various settings. It is possible to design a customized interface to improve the quality of the final visualization and to provide a better experience for users.

The project results can be used for other purposes. It can be used for educational or business objectives, and there is the capability to extend this platform in the future by interested students, which is not limited to offshore wind farms but can be extended to more applications and projects. Through its visualization feature, it allows users to explore and analyze the potential of real data and simulated data in an easy and efficient way, utilizing technologies such as machine learning and artificial intelligence for better decision-making and making it more agile.

A Appendix

A.1 Project repository

- **The Augmented reality of the proposed project is available for Android versions and can be downloaded through the following QR code.**

  How it works:

  1- Install the apk package on your Android phone and accept all the security warnings. Don't worry about it! ;)

  2- Open the app and focus the camera on the image target which is NTNU logo shown below. You can focus your camera wherever the NTNU logo can be located. But it should be almost similar the image target.

  3- Now the scene is available and ready for interaction. Have fun! :)

  Scan or click to download the app:

  Focus the camera on image target :

  NTNU

- **The project repository can be found and downloaded on GitHub:**

  GitHub

- **The Project video demonstrations and pictures can be found in the following shared google drive:**

  Google Drive

Bibliography

[1] W. Garlick, R. Dixon and S. Watson, 'A model-based approach to wind turbine condition monitoring using scada data', Jan. 2009.

[2] A. Rasheed, O. San and T. Kvamsdal. 'Digital twin: Values, challenges and enablers'. (Oct. 2019), [Online]. Available: https://www.researchgate.net/publication/336304126_Digital_Twin_Values_Challenges_and_Enablers.

[3] B. TROSKIEn. 'Most common reasons for wind turbine failures'. (), [Online]. Available: https://www.cotes.com/blog/most-common-reasons-for-wind-turbine-failures.

[4] P. J. Jordaens. 'Ten most common wind turbine failure issues identified'. (Nov 22, 2021), [Online]. Available: https://www.sirris.be/ten-most-common-wind-turbine-failure-issues-identified.

[5] S. Mein. 'Top 3 types of wind turbine failure'. (May 11, 2020), [Online]. Available: https://www.firetrace.com/fire-protection-blog/wind-turbine-failure.

[6] A. R. Aikin. 'Bearing and gearbox failures: Challenge to wind turbines'. (Jul 30, 2020), [Online]. Available: https://energycentral.com/news/bearing-and-gearbox-failures-challenge-wind-turbines.

[7] S. Dong, S. Yin, B. Tang, L. Chen and T. Luo, 'Bearing degradation process prediction based on the support vector machine and markov model', *Shock and Vibration*, vol. 2014, pp. 1–15, Mar. 2014. DOI: 10.1155/2014/717465.

[8] L. Saidi, J. Ben Ali, E. Bechhoefer and M. Benbouzid, 'Wind turbine high-speed shaft bearings health prognosis through a spectral kurtosis-derived indices and svr', *Applied Acoustics*, vol. 120, pp. 1–8, May 2017. DOI: 10.1016/j.apacoust.2017.01.005.

[9] Á. Encalada-Dávila, B. Puruncajas, C. Tutivén and Y. Vidal, 'Wind turbine main bearing fault prognosis based solely on scada data', *Sensors*, vol. 21, no. 6, 2021, ISSN: 1424-8220. DOI: 10.3390/s21062228. [Online]. Available: https://www.mdpi.com/1424-8220/21/6/2228.

[10] D. wind energy. 'Extracting power from the wind for a renewable future'. (2021), [Online]. Available: https://windenergy.dtu.dk/english.

[11] S. Boschert and R. Rosen, 'Digital twin—the simulation aspect', in *Mechatronic Futures: Challenges and Solutions for Mechatronic Systems and their Designers*, P. Hehenberger and D. Bradley, Eds. Cham: Springer International Publishing, 2016, pp. 59–74, ISBN: 978-3-319-32156-1. DOI: 10.1007/978-3-319-32156-1_5. [Online]. Available: https://doi.org/10.1007/978-3-319-32156-1_5.

[12] A. Fuller, Z. Fan, C. Day and C. Barlow, 'Digital twin: Enabling technologies, challenges and open research', *IEEE Access*, vol. 8, pp. 108 952–108 971, 2020. DOI: 10.1109/ACCESS.2020.2998358.

[13] dr. WJJ (Willem Jan) Knibbe et al. 'Kick-off digital twins: Let's find answers together'. (JUNE 19, 2019), [Online]. Available: https://www.wur.nl/nl/activiteit/Kick-off-Digital-Twins-Lets-find-answers-together-2.htm.

[14] E. Glaessgen and D. Stargel, 'The digital twin paradigm for future nasa and us air force vehicles', in *53rd AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference 20th AIAA/ASME/AHS adaptive structures conference 14th AIAA*, 2012, p. 1818.

[15] C. Verdouw, A. J. Beulens, H. A. Reijers and J. G. van der Vorst, 'A control model for object virtualization in supply chain management', *Computers in industry*, vol. 68, pp. 116–131, 2015.

[16] Oracle. 'A comprehensive approach to implementing iot digital twins'. (2019), [Online]. Available: http://www.oracle.com/us/solutions/internetofthings/digitaltwins-for-iot-apps-wp-3491953.pdf.

[17] Y. Arnautova. 'If you build products, you should be using digital twins'. (2022), [Online]. Available: https://www.globallogic.com/insights/blogs/if-you-build-products-you-should-be-using-digital-twins/.

[18] T. Thomas Plank CEO. 'Digital twins: The 4 types and their characteristics'. (2019), [Online]. Available: https://tributech.io/blog/the-4-types-of-digital-twins.

[19] C. Verdouw, B. Tekinerdogan, A. Beulens and S. Wolfert, 'Digital twins in smart farming', *Agricultural Systems*, vol. 189, p. 103 046, 2021, ISSN: 0308-521X. DOI: https://doi.org/10.1016/j.agsy.2020.103046. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0308521X20309070.

[20] J. Hagerty, 'Planning guide for data and analytics', *Gartner Inc*, p. 13, 2017.

[21] M. E. Porter and J. E. Heppelmann, 'How smart connected objects are transforming competition', *Harvard Business Review*, vol. 92, no. 11, pp. 64–88, 2014.

[22] C. N. Verdouw, J. Wolfert, A. Beulens and A. Rialland, 'Virtualization of food supply chains with the internet of things', *Journal of Food Engineering*, vol. 176, pp. 128–136, 2016.

[23] M. Grieves and J. Vickers, 'Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems', in *Transdisciplinary perspectives on complex systems*, Springer, 2017, pp. 85–113.

[24] N. Mohamed, J. Al-Jaroodi and S. Lazarova-Molnar, 'Leveraging the capabilities of industry 4.0 for improving energy efficiency in smart factories', *Ieee Access*, vol. 7, pp. 18 008–18 020, 2019.

[25] B. Dong, Z. O'Neill and Z. Li, 'A bim-enabled information infrastructure for building energy fault detection and diagnostics', *Automation in Construction*, vol. 44, pp. 197–211, 2014.

[26] R. Alonso, M. Borras, R. H. Koppelaar, A. Lodigiani, E. Loscos and E. Yöntem, 'Sphere: Bim digital twin platform', in *Multidisciplinary Digital Publishing Institute Proceedings*, vol. 20, 2019, p. 9.

[27] N. Mohammadi and J. Taylor, 'Knowledge discovery in smart city digital twins', in *Proceedings of the 53rd Hawaii international conference on system sciences*, 2020.

[28] B. Besselink, V. Turri, S. H. Van De Hoef *et al.*, 'Cyber–physical control of road freight transport', *Proceedings of the IEEE*, vol. 104, no. 5, pp. 1128–1141, 2016.

[29] J.-L. Vinot, C. Letondal, R. Lesbordes, S. Chatty, S. Conversy and C. Hurter, 'Tangible augmented reality for air traffic control', *interactions*, vol. 21, no. 4, pp. 54–57, 2014.

[30] L. Kent, C. Snider and B. Hicks, 'Early stage digital-physical twinning to engage citizens with city planning and design', in *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, IEEE, 2019, pp. 1014–1015.

[31] F. Tao, H. Zhang, A. Liu and A. Y. Nee, 'Digital twin in industry: State-of-the-art', *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2405–2415, 2018.

[32] T. Blochwitz, M. Otter, M. Arnold *et al.*, 'The functional mockup interface for tool independent exchange of simulation models', in *Proceedings of the 8th international Modelica conference*, Linköping University Press, 2011, pp. 105–114.

[33] E. Negri, L. Fumagalli, C. Cimino and M. Macchi, 'Fmu-supported simulation for cps digital twin', *Procedia manufacturing*, vol. 28, pp. 201–206, 2019.

[34] A. Sabir, N. A. Abbasi and M. N. Islam, 'An electronic data management and analysis application for abet accreditation', *arXiv preprint arXiv:1901.05845*, 2018.

[35] H. Laaki, Y. Miche and K. Tammi, 'Prototyping a digital twin for real time remote control over mobile networks: Application of remote surgery', *Ieee Access*, vol. 7, pp. 20 325–20 336, 2019.

[36] I. Fernandez-Ruiz, 'Computer modelling to personalize bioengineered heart valves', *Nature Reviews Cardiology*, vol. 15, no. 8, pp. 440–441, 2018.

[37] B. Zhang, A. Korolj, B. F. L. Lai and M. Radisic, 'Advances in organ-on-a-chip engineering', *Nature Reviews Materials*, vol. 3, no. 8, pp. 257–278, 2018.

[38] A. Kusiak, Z. Zhang and A. Verma, 'Prediction, operations, and condition monitoring in wind energy', *Energy*, vol. 60, pp. 1–12, 2013, ISSN: 0360-5442. DOI: https://doi.org/10.1016/j.energy.2013.07.051. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544213006579.

[39] C. Team. 'What are the statistical forecasting methods'. (May 12, 2021), [Online]. Available: https://collectivei.com/blog/what-are-the-statistical-forecasting-methods/.

[40] intellipaat. 'What is lstm? introduction to long short term memory'. (Feb, 2022), [Online]. Available: https://intellipaat.com/blog/what-is-lstm/.

[41] javatpoint. 'Support vector machine algorithm'. (), [Online]. Available: https://www.javatpoint.com/machine-learning-support-vector-machine-algorithm.

[42] K. Kutzkov. 'Arima vs prophet vs lstm for time series prediction'. (May 13th, 2022), [Online]. Available: https://neptune.ai/blog/arima-vs-prophet-vs-lstm.

[43] W. Badr. 'Auto-encoder: What is it? and what is it used for?' (Apr 22, 2019), [Online]. Available: https://towardsdatascience.com/auto-encoder-what-is-it-and-what-is-it-used-for-part-1-3e5c6f017726.

[44] Lightgbm. 'Lightgbm's documentation!' (), [Online]. Available: https://lightgbm.readthedocs.io/en/latest/.

[45] S. J. T. Ben Letham. 'Prophet: Forecasting at scale'. (Feb 23, 2017), [Online]. Available: https://research.facebook.com/blog/2017/02/prophet-forecasting-at-scale/.

[46] S. J. Taylor and B. Letham, 'Forecasting at scale', *PeerJ Preprints*, vol. 5, e3190v2, Sep. 2017, ISSN: 2167-9843. DOI: 10.7287/peerj.preprints.3190v2. [Online]. Available: https://doi.org/10.7287/peerj.preprints.3190v2.

[47] M. Krieger. 'Time series analysis with facebook prophet: How it works and how to use it'. (Feb 20, 2021), [Online]. Available: https://towardsdatascience.com/time-series-analysis-with-facebook-prophet-how-it-works-and-how-to-use-it-f15ecf2c0e3a.

[48] P. T. K. C. N. T. Wind. 'Digital twin for improved management of wind farms'. (), [Online]. Available: https://www.sintef.no/globalassets/project/industry-meets-science/22sept2020/presentations/trond-kvamsdal-ntnu-digital-twin.pdf.

[49] O. Oñederra, F. J. Asensio, P. Eguia, E. Perea, A. Pujana and L. Martinez, 'Mv cable modeling for application in the digital twin of a windfarm', in *2019 International Conference on Clean Electrical Power (ICCEP)*, 2019, pp. 617–622. DOI: 10.1109/ICCEP.2019.8890166.

[50] K. Sivalingam, M. Sepulveda, M. Spring and P. Davies, 'A review and methodology development for remaining useful life prediction of offshore fixed and floating wind turbine power converter with digital twin technology perspective', in *2018 2nd International Conference on Green Energy and Applications (ICGEA)*, 2018, pp. 197–204. DOI: 10.1109/ICGEA.2018.8356292.

[51] M. Wang, C. Wang, A. Hnydiuk-Stefan, S. Feng, I. Atilla and Z. Li, 'Recent progress on reliability analysis of offshore wind turbine support structures considering digital twin solutions', *Ocean Engineering*, vol. 232, p. 109 168, 2021, ISSN: 0029-8018. DOI: https://doi.org/10.1016/j.oceaneng.2021.109168. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0029801821006016.

[52] M. Botz, M. Raith, A. Emiroglu, B. Wondra and C. Grosse, 'Structural health monitoring as a tool for smart maintenance of wind turbines', in *Advances in Engineering Materials, Structures and Systems: Innovations, Mechanics and Applications*, CRC Press, 2019, pp. 1971–1975.

[53] J. D. M. De Kooning, K. Stockman, J. De Maeyer, A. Jarquin-Laguna and L. Vandevelde, 'Digital twins for wind energy conversion systems: A literature review of potential modelling techniques focused on model fidelity and computational load', *Processes*, vol. 9, no. 12, 2021, ISSN: 2227-9717. DOI: 10.3390/pr9122224. [Online]. Available: https://www.mdpi.com/2227-9717/9/12/2224.

[54] F. Pimenta, J. Pacheco, C. M. Branco, C. M. Teixeira and F. Magalhães, 'Development of a digital twin of an onshore wind turbine using monitoring data', *Journal of Physics: Conference Series*, vol. 1618, no. 2, p. 022 065, Sep. 2020. DOI: 10.1088/1742-6596/1618/2/022065. [Online]. Available: https://doi.org/10.1088/1742-6596/1618/2/022065.

[55] Vestas. 'Vestas wind energy solution'. (), [Online]. Available: https://www.vestas.com/en.

[56] Scipher. 'Scipher product'. (), [Online]. Available: https://www.utopusinsights.com/products.

[57] Siemens. 'Siplus cms – stepping up your production'. (), [Online]. Available: https://new.siemens.com/global/en/products/automation/products-for-specific-requirements/siplus-cms.html.

[58] ——, 'Mindsphere'. (), [Online]. Available: https://siemens.mindsphere.io/en.

[59] ——, 'Cms x-tools'. (), [Online]. Available: https://www.automation.siemens.com/bilddb/search.aspx?lang=de&mlfb=6AT8002-1AA00&useStructure=true&NodeID=1000000.

[60] GE. 'What is predix platform?' (), [Online]. Available: https://www.ge.com/digital/iiot-platform.

[61] S. Orhan, N. Aktürk and V. Çelik, 'Vibration monitoring for defect diagnosis of rolling element bearings as a predictive maintenance tool: Comprehensive case studies', *NDT and E International*, vol. 39, no. 4, pp. 293–298, 2006, ISSN: 0963-8695. DOI: https://doi.org/10.1016/j.ndteint.2005.08.008. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0963869505001349.

[62] I. E. Alguindigue, A. Loskiewicz-Buczak and R. E. Uhrig, 'Monitoring and diagnosis of rolling element bearings using artificial neural networks', *IEEE transactions on industrial electronics*, vol. 40, no. 2, pp. 209–217, 1993.

[63] M. MARTIN. 'How temperature conditions affect bearing performance'. (JUNE 10, 2016), [Online]. Available: https://www.bearingtips.com/temperature-conditions-affect-bearing-performance/.

[64] Y. Z. F. Xiao X.; Liu J.; Liu D.; Tang, 'Turbine main bearing based on multivariate time series forecasting', 2022. DOI: https://doi.org/10.3390/en15051951.

[65] I. Corporation. 'Vibration technical guide'. (), [Online]. Available: https://www.imv.co.jp/e/pr/vibration_measuring/chapter03/.

[66] L. D. Colomer, 'Deep learning for predictive maintenance of rolling bearings', M.S. thesis, Universitat de barcelona, July 1, 2020.

[67] N. C. Jonathan Trout. 'Vibration analysis explained'. (), [Online]. Available: https://www.reliableplant.com/vibration-analysis-31569.

[68] M. Singh. 'Why do we care so much about noramlity ?' (), [Online]. Available: https://github.com/manvendra7/Skewness-and-kurtosis.

[69] Equinor. 'Hywind tampen floating wind power project'. (), [Online]. Available: https://www.equinor.com/en/what-we-do/hywind-tampen.html.

[70] A. Olufemi. 'By abiodun olufemi'. (2020), [Online]. Available: https://www.academia.edu/4782531/wrap-fn.

[71] A. E. G. Data. 'Turbine scada dataset'. (), [Online]. Available: https://anero.id/energy/data.

[72] S. Martin del Campo Barraza, F. Sandin and D. Strömbergsson, *Dataset concerning the vibration signals from wind turbines in northern sweden*, 2018.

[73] NASA. 'Nasa pcoe datasets'. (), [Online]. Available: https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/.

[74] A. Haghshenas. 'Project repository'. (), [Online]. Available: https://drive.google.com/drive/u/0/folders/1lekG38P0UBOFLHjeLSd0IDXewk7T9KD3.

[75] opc-router. 'What is opc ua? a practical introduction'. (), [Online]. Available: https://www.opc-router.com/what-is-opc-ua/.

[76] fmi-standard. '150+ tools support fmi'. (), [Online]. Available: https://fmi-standard.org/tools/.

[77] C. Gomes, T. Blochwitz, C. Bertsch *et al.*, 'The fmi 3.0 standard interface for clocked and scheduled simulations', pp. 27–36, 2021.

[78] S. BHASKARPANDIT. 'Wind power forecasting'. (), [Online]. Available: https://www.kaggle.com/datasets/theforcecoder/wind-power-forecasting.

[79] eumetrain.org. 'Mean absolute error (mae) and root mean squared error (rmse)'. (), [Online]. Available: http://www.eumetrain.org/data/4/451/english/msg/ver_cont_var/uos3/uos3_ko1.htm.

[80] SimuTech. 'Ansysmodel'. (), [Online]. Available: https://simutechgroup.com/export-the-deformed-geometry-shape-from-an-ansys-model/.

[81] omnicalculator. 'Wind turbine calculator'. (), [Online]. Available: https://www.omnicalculator.com/ecology/wind-turbine.

[82] Mathworks. 'Matlabtcp'. (), [Online]. Available: https://se.mathworks.com/help/slrealtime/ug/target-to-host-communication-using-tcp.html.

[83] ——, 'Matlabudp'. (), [Online]. Available: https://se.mathworks.com/help/instrument/udp.html?searchHighlight=UDP&s_tid=srchtitle_UDP_1.

NTNU

Norwegian University of
Science and Technology