ICIT 2017 The 8th International Conference on Information Technology
doi:………….. © ICIT 2017 (http://icit.zuj.edu.jo/ICIT17)

A Model for Adopting Sustainable Practices in Software Based Organizations

Bokolo Anthony Jnr.
Faculty of Computer Systems and Software Engineering
Universiti Malaysia Pahang (UMP)
Lebuhraya Tun Razak, 26300 Gambang, Pahang, Malaysia
bkanjr@gmail.com

Mazlina Abdul Majid
Faculty of Computer Systems and Software Engineering
Universiti Malaysia Pahang (UMP)
Lebuhraya Tun Razak, 26300 Gambang, Pahang, Malaysia
mazlina@ump.edu.my

Awanis Romli
Faculty of Computer Systems and Software Engineering
Universiti Malaysia Pahang (UMP)
Lebuhraya Tun Razak, 26300 Gambang, Pahang, Malaysia
awanis@ump.edu.my

***Abstract*—The present need for software organizations to seamlessly carryout software development objectives and aims results to significant high cost of electricity usage and the emission of CO_2 in the environment and climate. Therefore there is need for software practitioners to adopt sustainable practices in software development process, when they utilized software to lessen electricity usage and reduce CO_2 emission. Sustainable practice adoption is an attempt for software practitioners to tackle the current climatic problem, and can also improve the economic performance of software organizations. Software practitioners' adoption of sustainable practices is beneficial to the society, climate and humanity. Existing approaches mainly focus on achieving sustainability when software practitioners develop software. Thus this paper contributes to sustainability research by presenting a model that offers an understanding of how software practitioners can adopt sustainable practices when they utilize software applications in software process. Findings from this paper show how Green practices can be diffused by software practitioners when they utilize software application in software organizations.**

Keywords—Sustainability; Software Practitioners; Software Organisation; Eco-Environment; Software Usage

I. INTRODUCTION

Sustainability has been comprehensively discussed within corporate management based on the synonyms of corporate social responsibility (CSR). The word sustain was derived from the Latin "sustenere" meaning contributing to sustainable development by concurrently delivering social, economic and environmental benefit [1], [2]. In software process it can be termed as survival guarantee, meaning that an ecological, economic or social system should be preserved for future generations. Thus necessary resources should only be utilized to a degree where it is possible to restore them within a rejuvenation cycle [3]. As software industries increases the call for sustainable practices more pressure is put on software practitioners to comply. The question of software organizations adopting Green practices is no longer whether or not they will, but when will they. As government and public advocates increase their call for environmental sustainability, more and more software industries such as Dell and IBM are implementing Green strategies [4]. The quest of sustainability, which is mostly defined as meeting the needs of the present generation without compromising the ability of future generations to meet their needs, is an essential global issue. There are three dimensions to sustainability as stated previously economic, social, and environmental commonly known as the triple bottom line. Software organizations need to develop sustainability approaches that balance economic gains, societal, and environmental goals in such a way that all three

doi:………….. © ICIT 2017 (http://icit.zuj.edu.jo/ICIT17)

are considered instantaneously. This research primarily addresses the concern of eco-sustainability which requires that software practitioners should take cooperative action to tackle environmental problems and inverse or reduce the effects of global climate change as it relates to software usage in software process [5]. Sustainability has thus become one of the latest thinking on improving climatic and eco-environment goals while reducing the cost of software processes. Bokolo and Noraini [6] states that sustainability focused mainly on energy efficiency and use of equipment, aimed at minimizing the expense generated from information technology (IT) usage.

Sustainability in software process is the practice of developing, use and disposal of computers, servers and related subsystems such as monitors, printers, storage devices, and networking and communications systems efficiently and effectively with minimal or no impact on the environment. Software practitioners can adopt sustainable practices in software process to ensure efficiency of services, reduced costs, and better control over system infrastructure [7]. Therefore sustainable practice can reduce costs, resulting in savings revenue. Sustainable software practice is related to the use, deployment and implementation of software systems that contribute to sustainable processes.

Watson, Boudreau, Chen and Huber [8] stated that sustainable practices in software organisation should consist of the implementation of policies for the use of IT for emission control, energy, and other assets of the organisation, thus increasing energy efficiency and management through the transformative power of sustainable implementation in software based organisation. For this reason software practitioners can benefit from the adoption of sustainable business practices, optimize their financial and economic performance as a result of lower costs derived from eco-environmental practices [9].

Therefore sustainability in software process should include all about data center efficiency, reduce power and cooling technology initiatives to reduce costs associated with data center operations. Sustainability should also consider eco-environmental practices in the acquisition, use and effective disposal of e-wastes [10]. Therefore this research paper presents a model that supports software practitioners in software based organisation diffuse sustainable practices in their software process when they use software applications and other related infrastructures such as database server, communication server and peripherals. The proposed model can be adopted by software practitioners as benchmark model to assess their sustainable pursuit.

The outline of this paper is organized as follows: Section 2 presents the materials and methods. Section 3 describes system infrastructure in software organisation. Section 4 is the proposed model. Section 5 is the discussion. Final section is the conclusion, limitation and future work.

II. Materials and Methods

To gain insight on the diffusion of sustainable practices in software organizations, a review on sustainable and eco-environmental practices was carried out. The reviewed literatures assisted in developing a theoretical and conceptual understanding of sustainability in software organizations in developing a model to assist software practitioners in adopting sustainable practices when they utilize software applications in software processes.

A. Sustainability in Software Organizations

Sustainability is a concept of preserving the long-term well-being for humans. Sustainability is the ability of one or more objects, or entities, either individually or together, either be affected or altered forms for a long time.

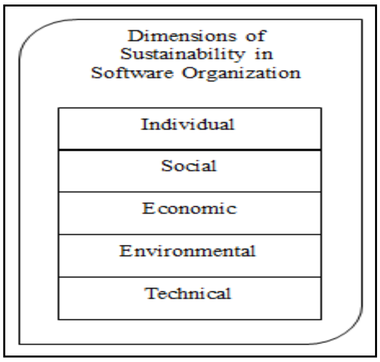

Fig. 1. Dimension of sustainability in software organizations [13]

Sustainability aims to help software organisation diffuse Green practices [11]. Katrina, Anni-Kaisa, and Paavo [12] stated that sustainability is the ability to reduce the long-term effects associated with fluctuations in the cost of energy, resource depletion and pollution, liability products, and waste management in organisation, thus ensuring that dangerous substances are not used in the production process in software organizations. Sustainability is categorized into 5 sections in software organisation, as shown in Fig 1.

Thus Fig 1 shows the five dimensions are to be considered in implementing sustainability in software organizations, they are described as follows; individual sustainability refers to sustaining software practitioner resources based on their health, leadership, education, knowledge, skills, and access to services. Social sustainability aims at sustaining the societal communities in their services and shared aims. Economic sustainability aims at sustaining capital and adding value. Environmental sustainability refers to improving nature and climate by protecting the natural resources such as land, water, minerals, air and eco-environmental services. Technical sustainability refers to longevity of software infrastructure systems. Technical sustainability also supports better establishing of eco-environmental concerns with respect to enterprise current practice.

Thus software organizations must pay close attention to climatic changes and environmental protection, as the software industries uses resources such as water, energy and hazardous materials in development process. The current software processes generate waste that can harm the environment and results to more climatic changes; therefore, it is important to software practitioners to adhere to environmental policies issued by government and non-governmental institutions. That is why sustainability is becoming a program that enlightens software practitioners to develop, manufacture and use of computers, servers and peripherals efficiently and effectively reduce damage to the environment [7].

B. Strategic Alignment to Sustainable Practice

In software organizations a corporate strategy is implemented to guide software practitioners' actions and set directions for eco-environmental sustainability. Hence to attain eco-environmental sustainability, software practitioners need to incorporate sustainability into software processes.

Fig. 2. Strategic alignment of sustainability

Based on work by Watson et al. [8] software strategy can be addressed from the perception of aggregation, adaptation and arbitrage. However there is a need for another strategy called articulation. This is a new strategic added to the existing "3" strategies mention previously by [8]. Fig 2 assist software practitioner to move faster and more efficiently thus creating sustainable software practices and gain a competitive advantage, thereby reducing risk to the environment thus eliminating e-waste. Aggregation aims to reduce costs by integrating Green actions for eco-efficiency [8]. Software systems can be used to reduce risks to the environment, such as high costs, emissions and waste from each process stage of software production in organisation. Aggregation is also a tool for aggregating and organizing sustainable business activity to reduce overall emissions. Virtualization can also be used as reduction techniques by combining different software systems on a single server to increase utilization and reduce costs.

Adaptation defines software organizations efforts to capitalize on its local significance by being responsive to end user needs. For eco-environmental sustainability of business, software practitioner needs to take environmental initiatives that reduce emissions and waste in their organisation [8]. Arbitration aims to achieve total savings through software practitioners' experiences and make decision on selecting the best alternatives considering the environment in each development process. This can be seen as achieving the most eco-friendly to the environment by choosing the most environmentally friendly merchants [8]. Articulation involves the plan and application of software that contribute to sustainable in enterprise process. Thus sustainability initiatives in enterprise process are mitigating environmental risk, there by achieving the organisation goals. This can be achieved through the reduction of pollution, product stewardship, or cleaner technology [1], [11].

C. Green Energy Metrics in Software Organizations

This sub section briefly explains the Green energy metric to be considered by software practitioners when they utilize software in software process.

TABLE I. GREEN ENERGY METRICS IN SOFTWARE ORGANIZATIONS[14]

Energy Metric	**Description**
Software Application	This metrics is concern with computer software programs or pieces of code. Application metric is measured by evaluating the energy cost due to input/output operations.
System Architecture	Mainly evaluates the energy cost due to CPU processing, memory access operations of any application. It is measured by estimating the energy utilized by software system.
Service	This metrics measure energy consumption generated by the execution, running and the deployment of software service. This metric is measured based the efficiently of a service energy usage.
IT Service Center	This metrics measure the effect of service implementation on a service center. This metrics can be measured by estimating the energy consumption produced by both active servers.
Virtual Machine	Aims at estimating or to assess energy consumption produced by virtual machines. Moreover, this metrics rates the effects of virtualization on energy consumption by measuring the disk over duration.
Enterprise Data Center	Comprises all those metrics that execute measurements of the impact of data storing and retrieval on enterprise data center. This metric measurement is carried out by estimating the number of bytes that are processed (useful work) per kWh of electric energy with respect to the whole data center.
Embedded Software	Involves metrics that are used to perform r evaluations on software that interacts directly with software practitioners. This metric evaluates the energy utilization in dealing with the number of deployed assembly commands and considering a distinctive embedded integer processor core.
Enterprise Server	Consist of metrics that perform estimates on the impact of application, data processing or service on a server machine. Measured by estimating the amount of power used by a server with respect to the server CPU
Knowledge Base	Describes all those metrics that are designed to rate the energy budget of data storing and retrieving operations. It is measured based on the energy required to carry out a query form the firms' knowledge base.

The metrics are shown in Table 1 are important to attain sustainability in software process, since each of these metrics consumes electricity. It is important for software practitioners to be aware of these metrics in their enterprise.

D. Related Works

A few studies has been carried out to investigate how software organizations can implement Green practices in software processes, mainly in software development process to reduce energy usage and reduce CO_2 emissions at the same time reduce cost incurred in software related process. One of these studies was research carried out by Brigit [13] who presented a checklist and guide word based method that shows how to include the aims and goals of environmental sustainability from the very initial steps in discovering the stakeholders and analyzing the area to the definition of a usage model and specific requirement. The researcher aims to offer a flawless perception on how environmental sustainability can be methodically supported as a goal in requirements engineering process.

Bokolo and Mazlina [15] contributes and studies on the use of information systems to support sustainability in software organizations by offering a preliminary insight into the

doi:………….. © ICIT 2017 (http://icit.zuj.edu.jo/ICIT17)

operation and implementation of sustainability in software management process as well as the variables that effect eco-environmental practices by recommending a model that will help decision making for sustainable software management process. The model comprises of pressure, software practitioners, software governance and IT infrastructure as independent variables, the current Green practices as mediating variable and Sustainability target as dependent variable.

Stefan et al. [16] developed a Greensoft model, a model based on achieving a Green and sustainable software development. The model assists software development process to decrease energy and resource usage, as well as the use of ICT to facilitate software development process. The model also provides a clear report on resource and energy savings through ICT through resource and energy consumption by ICT usage. Markus, et al. [17] designed a model that assimilates Green IT aspects into software engineering procedures with agile approach in order to produce Greener software. The researcher integrated an agile method to allow software development team to take over the obligation for project decisions that have a direct effect on the environmental and, concerning hardware requirements, also on the societal impacts of the software product.

Albert [18] proposes a revised systems development lifecycle (SDLC) that contemplates environmental issues at every stage of the system development process and recommends the addition of a disposal phase as an official, final stage of the systems development process. The researcher argues that recycling alone is insufficient to address the increasing environmental effect of information systems (IS) or information technology (IT) and that it is too late to resolve, let alone address the issue at the end of the product or service lifecycle. Thus environmental issues need to be considered throughout the lifecycle of the software development system, comprising practices regarding software, hardware and end users.

Mohankumar and Anand [19] developed a Green IT star model method to assist software development life cycle. The model can assist software developers when they develop and implement software. The model comprises of initial stage, repeatable stage, defined stage, managed stage and optimizing stage. Each of these stages is influenced by the requirement, design, coding, testing and implementation phase of the production process. Paolo et al. [14] carried out a systematic literature review on Green software metrics to be considered in software development process. These metrics are to be considered by software practitioners when they develop software. Additionally, their study created an understanding of how to assess the software Greenness for measuring the energy effectiveness in software lifecycle. The study also classified and described the software metrics associated to software Greenness.

Patricia, Rick and Niklaus [20] discussed on exploring initial challenges for Green software engineering by discussing that software can contribute to reduce energy consumption (i.e. become Greener) in at least two ways. First, by being more energy proficient, hence using fewer resources and causing less CO_2 emissions. Secondly, by making software process more sustainable, i.e. reducing the emissions of software companies. Juha [21] recommended a set of Green software quality factors (feasibility, efficiency, sustainability and reflectivity) to assist software organisation develop good and beautiful software and avoid developing bad software. Juha [21] mentioned that good software helps to lessen greenhouse gases, waste and resource requirement and while bad software increases them.

The reviewed Green software models or frameworks aims to assist software practitioners achieve sustainability when they develop software. However none of these reviewed model/frameworks aims to support software practitioners in achieving sustainable software organisation when they use software application to carry out software process.

III. System Infrastructure in Software Organization

This Section illustrates the structure of system infrastructure in a typical software based organisation as seen in Fig 3. In software organisation, software asset capture and process information whereas infrastructure referred to the more inclusive mixture of IT as seen in Fig 3.

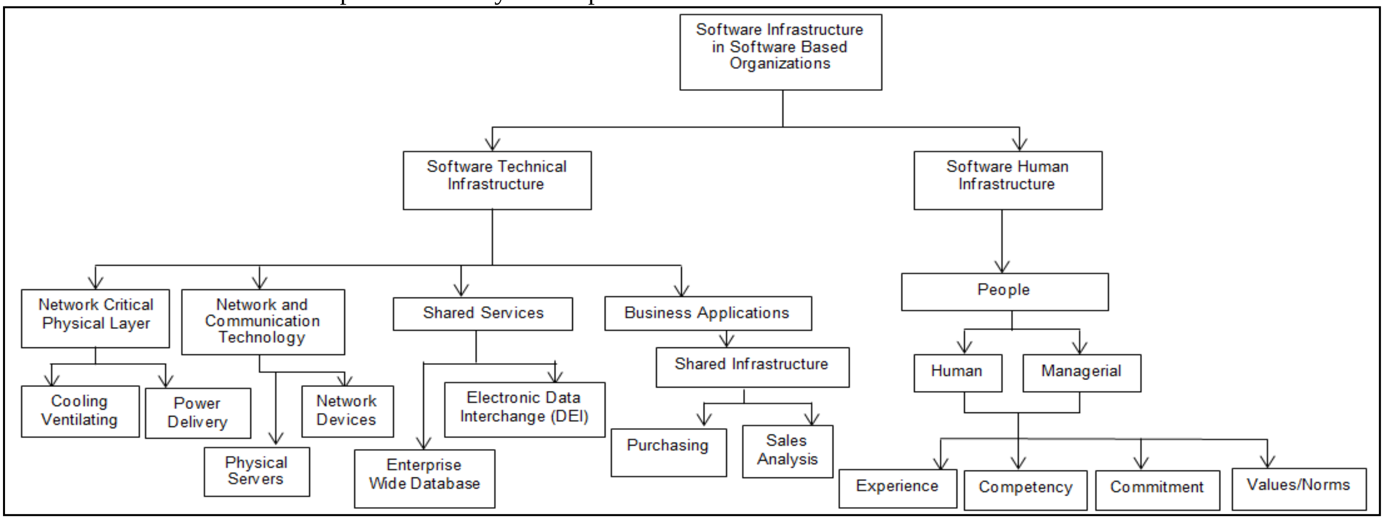

Fig. 3. System infrastructures in software based organisation [22]

doi:………….. © ICIT 2017 (http://icit.zuj.edu.jo/ICIT17)

Fig 3 shows the system infrastructure of an organisation comprising of the technical software infrastructure and the software human and managerial capability infrastructure. The Software human infrastructure relates to the commitments, competencies, experiences, values and norms of software practitioners carrying out software process.

The managerial capability includes the management and decision makers in the organisation [23], [24]. Stan et al. [25] suggested that to improve energy efficiency in software organisation four basic choices need to be diffused by software practitioners; the first is to lessen computer power usage through efficient software management; second option is to improve server efficacy; third step is to improve the proficiency of power distribution and supplies; and lastly software practitioners need to improve the proficiency of cooling systems.

Software technical infrastructure is theorized as a structure of four layers: the network critical physical infrastructure such as in ventilating, power delivery and cooling; Software network and communications technologies as seen in network devices and physical servers; shared services mainly in electronic data interchange and organizational databases. Business applications that utilize the shared infrastructure in sourcing analysis, purchasing.

Therefore sustainability in software organisation for eco-environment is a methodical application of ecological-sustainability benchmarks in the process of software technical infrastructure as well as within the human and managerial dimensions of system infrastructure in order to reduce CO_2 emissions, water use and waste; but rather generate Green financial returns and improve energy efficiency.

Alemayehu [24] suggested that IT can provide some of the best tools not only in measuring and reporting CO_2 emissions; waste and water use within software organizational processes but can also assist in reducing environmental issues.

A. Factors that Influences Sustainable Practice

Watson et al. [8] mentioned that IT is both a cause of the environmental issues and also a potential solution to eco-environmental sustainability. In his own view IT equipment such as coal-fired power plants providing electricity it needs to support software processes, but also generates CO_2 that results to global warming and climatic changes.

However, with the use of renewable energy technologies in software process, based on wind and solar are possible solutions for achieving sustainability in software organizations.

TABLE II. DRIVERS THAT INFLUENCE SUSTAINABILITY

Forces	Description
Economic Factors	Refers to the necessity of attaining internal efficiency and market routine and could be major promoters of sustainability. It related to the need for reducing the power, cooling and property costs and enhancing data center efficiency might induce software organisation to turn to sustainable practice [23]. Thus economic forces relates to greater IT efficiency and the search of tangible cost savings from IT operations [26].
Regulatory Factors	Are important in checking if software organizations adhere to eco-environmental responsible behaviors or initiatives [24]. Inter-governmental organizations could enforce software organizations to adoption sustainable rules, regulations and laws that prohibit certain enterprise practices. These included banning the use of hazardous materials and imposing restrictions on e-waste disposal [26].
Normative Factors	Refer to the search of acceptability within the wider public context. The need to meet public obligations and enforce moral ethics can influence certain sustainable practices. Presently a number of professional, national, and intergovernmental institutions are producing strategies and standards related to sustainability. Such ingenuities create the systems for self-regulation and can lead to sustainable practices that align enterprise to socially accepted sustainable behaviors such as reuse and electronic waste management, reducing emission and recycling [23], [24], [26].

However, the adoption of sustainable practices can be influenced by a number of drivers as described in Table 2. Sustainable approach supports software organisation achieved its objectives and procedures for organic Greener approach there by reducing the risk to the environment. So sustainability should not be seen as a cost of doing business. Rather, it is opportunities for software organisation to improve the productivity of services reduce costs and increase profitability. Poor software practice result in many forms of environmental risks and these risks results to energy waste, unused funds, heat, noise and emissions. Sustainable practices can lead to less electronic waste, reducing cost incurred thus caring for the environment, society and people.

IV. PROPOSED MODEL

This section explains how to adopt sustainable practice in software based organizations.

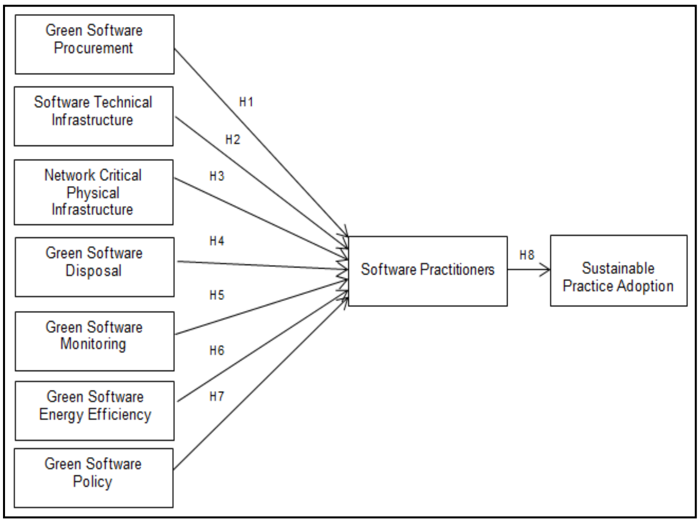

Fig. 4. A model for adopting sustainable practices in software organizations

Software practitioners can measure and estimate the adoption of sustainability practice based on the independent variables, mediating variable and dependent variable as shown in Fig 4. This section also explains on the results on how to implement sustainable in software organizations. Software practitioners can measure and estimate implementation of

Green sustainability practice based on the independent variables and mediating variables shown in Fig 4. These variables in Table 3 and Fig 4 can be seen as independent variables (Software Technical Infrastructure, Network Critical Physical Infrastructure, Green Software Disposal, Green Software Monitoring, Green Software Energy Efficiency, Green Software Procurement and Green Software Policy) that are mediated by the mediating variable (Software practitioners) that influence the dependent variable which is "sustainable practice adoption".

The developed model in Fig 4 is based on previous work by [10], [22], [23]. The model identifies the variables to be considered by software practitioners in diffusing sustainable practices in their organizational process when they utilize software. Thus the variables can influences climatic changes by reducing the amount of CO_2 in the environment, software practitioners need to consider these variables in software process when utilizing software application. Fig 4 shows the proposed model for sustainable practice adoption in software based organisation. The model support software practitioners to adopt sustainable software process when they utilize software application to carryout software procedures. The independent variables (Green software procurement, software technical infrastructure, network critical physical infrastructure, Green software disposal, Green software monitoring, Green software energy efficiency and Green software policy) are mediated by the mediating variable "software practitioners", which in turn influences the dependent variable "sustainable practice adoption". In the subsequent Section this research paper proceeds to discuss on the model variables. The independent variable, mediating variable and dependent variables are explained. Also the model hypotheses are derived to support the correlation between the (IV) independent variables, (MV) mediating variable and (DV) dependent variable.

TABLE III. VARIABLES AND GREEN STRATEGIES FOR SUSTAINABLE PRACTICE ADOPTION [3], [6], [28], [29], [30], [31], [32]

Variables	Sustainable Practice Adoption Strategies (Items)
Software Technical Infrastructure	-Server consolidation Storage/desktop virtualization. -Data de-duplication rightsizing software equipment, storage tiering. -Install disaster recovery software application. -Shift to cloud computing technologies and practice grid computing. -Manage and control lifecycle of stored data to avoid data redundancy. -Use knowledge management tools and carryout data de-duplication. -Proper software e-documentation practice. -Efficient coding when using software application. -Reuse of software modules and knowledge when coding.
Network Critical Physical Infrastructure	-Deploy teleconferencing technology. -Utilize video conferencing in software process. -Encourage telecommuting among software practitioners. -Utilize more of online collaboration infrastructures. -Install more energy efficient lights and efficacy stand-by power systems. -Implementation of energy efficient communication and database servers. -Remove unused communication and database servers. -Deploy intelligent switches. -Connect network multifunction printers.
Green Software Disposal	-Disposes of software in an environmentally friendly manner. -Recycles software consumable equipment such as ink cartridges, batteries and paper. -Print double sided on paper. -Adhere to environmental regulations on dumping e-waste. -Practice e-waste management if possible. -Reuse old computers systems and refurbish old computers systems. -Prolong life of IT/software equipment and recycle computer related hardware. -Keep track of toxic materials generated form software process. -Implement E-waste strategies and rules, if possible. -Involve in recycling initiatives. -Cooperate with merchants and advocate for take back programs and recycling practices.
Green Software Monitoring	-Develop a Green software action plan. -Develop a Green software service and product portfolio. -Implements CO_2 footprint monitor. -Analyses software's energy bill separately from the overall commercial bill shortens software refresh periods to access energy efficient equipment. -Audits the power efficiency of existing software systems. -Assessing the environmental effect of software usage. -Install airflow and liquid cooling monitoring system. -Implement thorough monitoring of air temperatures in the enterprise data center. -Enhance air flows and reuse enterprise data center heat.
Green Software Energy Efficiency	-Operates current software systems in an energy efficient manner. -Imposes personal computer (PC) power management. -Switches off data center lights and equipment when not in use. -Airside/waterside economizer, liquid cooling for software equipment. -Upgrades to more effective transformers. -Facilitate power supervision of all applicable system servers. -Permit data center level power management of networking, software applications and storage facilities. -Use more efficient transformers and uninterruptible power supply (UPS). -Install more efficient fans, chillers and pumps. -Practice free cooling or enterprise server by using proper ventilation from nature. -Deploy scheduling and workload management. -Configure server to shut down dynamically when not in use. -Deploy blade servers and energy-efficient processors. -Use energy-saving hard disk drives. -Use dynamically modifiable fans. -Lessen power conversion steps to decline power losses. -Use notebooks as a substitute of desktop computers. -Mount thin clients and utilize LED displays. -Relocation of its enterprise data center near clean/natural sources of energy.
Green Software Procurement	-Preference of software merchants that have a Green track record. -Prefers hardware vendors that offer take-back options. -Gives weight to environmental considerations in software procurement. -Purchase recycled IT equipment for software use. -Conduct lifecycle analyses (LCA) and total-cost-of-ownership (TCO) -Purchase eco-friendly cartridges and paper. -Buy renewable energy and energy proficient cloud services. -Check for certified merchants or vendors.

doi:………….. © ICIT 2017 (http://icit.zuj.edu.jo/ICIT17)

Variables	Sustainable Practice Adoption Strategies (Items)
Green Software Policy	-Environmentally friendly software purchasing policy. -Implement policy on software practitioners' use of software in an energy efficient manner. -Create policy on the usage of software to decrease CO_2 footprint. -Implement policy on handling electronic waste. -Adhere to government and non-governmental energy efficiency guidelines. -Allocate budget for Green practices in software related process in the organisation. -Outline environmental requirements for software vendors. -Collaborate with IT/Software infrastructure merchants and share knowledge. -Encourage merchants to lessen their footprint. -Carryout environmental merchant audits. -Consume and utilize electricity delivered by Green energy providers if available.
Software Practitioners	-Carryout Green training and education among software practitioners. -Codify existing software experts Green Knowledge and experience. -Create public awareness on Green practice in software process through workshops, colloquium and seminars. -Always print on both sides of a paper. -Participate in Green software practice discussions and forums. -Subscribe to Green software e-mail delivery lists. -Reuse all possible resources in software process. -Minimize input and if possible maximize software output. -Create lasting incentive systems by minimizing hazardous substances.

Table 3 shows the variables and strategies to support software practitioners in adopting a sustainable practice in software process. Sustainability in software organizations is the practice of manufacturing, utilizing, designing and disposing of servers, computers hardware and associated subsystems efficiently and effectively with minimal or no impact on the environment. The aim of sustainability is to achieve economic viability and improved social and environmental responsibilities ethically. Therefore Table 3 can be used by software practitioner to benchmark their existing software process against the variables and sustainable practice adoption in Table 3.

Furthermore, Table 3 provides an agenda for implementing eco-friendly practice in software organisation when software practitioners use software application. Table 3 also provides insights and guidelines for software practitioners, and sustainability managers to implement adequate sustainable eco-environmental practice in their software processes.

A. *Green Software Procurement*

This variable encourages software practitioners to purchase infrastructure that has Green label, low CO_2 emission or energy saver logo in this appliances. This is a sustainable practice that should be carried out by software practitioners. Green software procurement involves the practice of environmentally preferable IT infrastructure purchasing in software enterprise [7]. Green software procurement involves strategies such as assessment of the Green track record of software application, analysis of the environmental foot print of an IT hardware and inclusion of social concerns such as the presence of harmful materials in IT procedure in Green purchasing decisions [7]. Based on the Green software procurement variable, the hypothesis is formulated;

H1: Green software procurement practice will have a positive effect on the adoption of a sustainable software organisation.

B. *Software Technical Infrastructure*

This variable mainly comprises of the hardware and peripheral devices that the software are installed on and are used in software development process, this also contributes to climate change as such should be considered by software practitioners. Software technical infrastructures are mostly software systems utilized for decreasing the energy consumption of powering and cooling software infrastructure assets. Software technical infrastructure aims to optimize energy efficiency of software by lessening software induced greenhouse gas emissions [22], [27]. Based on the software technical infrastructure variable, the hypothesis is formulated;

H2: Software technical infrastructure has a positively effect on the adoption of a sustainable software organisation.

C. *Network Physical Techical Infrastructure*

Network critical physical infrastructure is based on the communication devices utilized by software practitioners for communicating when carrying out software process. The network critical physical infrastructure also controls the communication applications that assists software enterprise meet regulatory and supervisory reporting and certification requirements by managing the sustainable declaration processes more efficiently, and reduce environmental risk and cost reduction.

Network critical physical infrastructure helps software organizations ensure compliance with environmental laws and policies and reduce associated costs, efforts, and risks as stated previously that can be generated form communication infrastructures in the enterprise [23], [24]. It controls all eco-sustainable processes by unified incorporation with network operations data control and network control systems. Based on the Network critical physical infrastructure variable, the hypothesis is formulated;

H3: The adoption of a sustainable software organisation varies based on the network critical physical infrastructure utilized in their software process.

D. *Green Software Disposal*

This variable involves how software practitioner dispose of e-wastes (e wastes are waste derived from old hardware and network infrastructures that are longer useful or are outdated) this wastes should be reuse, recycled and refurbished by software firms/enterprise [6], [7], [11], [15], [22]. Thus Green software disposal refers to practices in recycling reusing and disposing IT hardware in an eco-friendly manner. Based on the Green software disposal variable, the hypothesis is formulated

H4: Green software adoption in software organisation will positively be successful if software are reuse, refurbish, recycle and disposed of properly.

doi:………….. © ICIT 2017 (http://icit.zuj.edu.jo/ICIT17)

E. Green Software Monitoring

This variable encourages software practitioners to check and report the information on the amount of CO_2 generated to the environment during software process, since CO_2 emission results to climatic changes which is bad to our environment as well-being [6]. Thus there is need for software to collect information related to the Greenhouse gas emission. Based on the Green software disposal variable, the hypothesis is formulated;

H5: The availability of Green software information monitoring in software organisation will positively affect the adoption of a sustainable software organisation.

F. Green Software Energy Efficiency

This variable comprises the usage of software infrastructure that utilizes less energy in software process, hence reducing the electricity bill incurred by the organisation, thus software practitioner need to be energy conscious when they develop software [6]. Thus there is need for software practitioners to practice and utilize software and hardware infrastructure that are energy efficient. Based on the Green software energy efficiency variable, the hypothesis is formulated;

H6: Energy efficient software and hardware usage by software practitioners will have a positive effect on the adoption of a sustainable software organisation.

G. Green Software Policy

This variable assess if the management in the organisation encourages sustainable practices or not. If the management encourages sustainable practices the software practitioners also practice Green strategies in software development process. According to Alemayehu et al. [22] policy concerns sustainable criteria and outlines software enterprise puts in place to guide the Greening of software process and software usage in relation to the activities carried out by software practitioners.

However, not all software policies are likely to be easily integrated into software process, nor are all software practices expected to be policy directed. Green software policy describes the magnitude to which sustainable issues are captured in software organizational processes [23]. Thus Green software policies encapsulates software organization‘s determination to go Green. Based on the Green software policy variable, the hypothesis is formulated;

H7: Green software policy will be positively related to the adoption of a sustainable software organisation.

H. Software Practitioners

This is the mediating variable that depends on all the other independent variables in the model. This variable utilizes the independent variables. Software practitioners are software directors, experts, software system designers and system developers. Software practitioners' obligation is mandatory in software development process because they are involved in scheduling, executing, deploying, testing and updating the system software with environmental sustainability considerations in mind [6].

Thus software practitioners cooperation is based on their common interest and incentives in accomplishing the aims or necessities of the industry, which is to implement a quality software in due time and at a lesser cost [11], [15], [27]. Also software practitioners are involved in using the software application to carry out software development tasks. Based on the software practitioners' variable, the hypothesis is formulated;

H8: Software practitioners' actions have a positive effect in the adoption of a sustainable software organisation.

I. Sustianable Practice Adoption

This is the dependent variable (DV) that is mediated by the mediating variable (MV) "software practitioners". Sustainable practice adoption variable is the current software development process carried out by software practitioners in software industries. Based on research by Stefan et al. [16] five main process are considered in software practice are considered in implementing and deploying Green sustainable software. They include development, distribution, operation, disposal and deactivation. These phases are carried out by software practitioners when they implement their software process using software application. Sustainable practice adoption in software organisation is shown in Table 3.

V. Discussion

Sustainability is defined as the ability to preserve or conserve a certain state or process indefinitely. In software process sustainability refers to the exploitation or consumption of natural resources that does adversely change the resources state beyond what is satisfactory. But presently, there are increasing concerns about the accelerating deterioration of natural resources and the environment. The United Nations General Assembly acknowledged that these environmental issues were universal in nature and resolved that it was in the common interest of all countries to integrate policies for a more sustainable development [33]. At the moment several governmental regulations are inducing leading software based organizations such as Microsoft Corporation; Sun Oracle etc. to review their enterprise practices by making them more sustainable. Hence, the attainment of sustainable practices is a key issue confronting software based organizations.

One of the most challenging issues faced by software based organization currently is the adoption of sustainable development in software processes, which is how to meet the needs of the present without conceding the ability of future generations to meet their own needs. As software practitioners/engineers, there is need to contribute to these goal by integrating sustainable practices to the present software analysis techniques and requirements modelling as a means to adopt sustainable requirements in organizational decision making. Nils-Holger et al. [3] stated that due to the increase in global impact of software usage on the society, ecology and the economy, software organizations are challenged to take the conception of sustainability into development process.

Sustainable development in software organizations can be stated as a thought of long-term concurrent optimization of ecological, economic and social goal to produce a long-term greater commercial performance for software organizations. Environmental problems derived from waste of electronic products (e-waste) are presently increasing as the year go by, making it the fastest rising waste in the industrialized sector. Waste generated form obsolete and outdated IT facilities such as monitors make up 25 % of electronic waste generated yearly. This shows that a least value of five million tons yearly of IT related wastes are being generated. It is imperative that software based organizations have no better option other than to deal adopt sustainability practices into day-to-day processes.

VI. Conclusion, Limitation and Future work

In the last few decades, the advance of software based industries has transformed most markets into e-economy and productions into e-business. Thus, software practitioners are increasingly playing serious roles in climatic changes in the world. However, global warming and climate change combining with inadequate availability and rising cost of energy are posing serious issues to the sustainability of the world. Sustainability can be achieved in software based organizations by adjusting existing software engineering techniques for reasoning and modelling to involve sustainability aspects within the organizational software activity. This can also be attained by integrating sustainability concerns in the present software development paradigms, aimed at lessening CO_2 emission, reducing energy usage, cost incurred decrease, eco- friendly e-waste disposal, water wastage decrease and preservation of natural environment.

Moreover, software practitioners (system designers, programmers and developers) should implement software systems in a more flexible way that supports “Greener” opportunities, i.e. allowing end users to make cost reduction options if they desire. Sustainability can be seen as an extra prerequisite software organization must also satisfy. Therefore there is need for an approach to facilitate communication between software practitioners and directs the comparison and consideration of different solutions based on qualitative information such as expert judgment rather than actual data [34]. According to Jordi et al. [34] there is also a need for a guide to support and allows software practitioners in viewing and defining sustainable actions.

This research paper answered the call from Jordi et al. [34] and also presented the guide in Table 3 by conducting a based on desk-based research utilizing secondary data source form previous works to develop a model. Thus this research paper contributes to sustainability research by presenting a model that offers an understanding of how sustainable practices can be diffused practically in software organizations when software practitioner uses software applications. The model can support the pursuit of sustainability in software based organisation by supporting software practitioners in adopting sustainable practice. The developed model comprises of independent variables (software technical infrastructure, network critical physical infrastructure, Green software disposal, Green software monitoring, Green software energy efficiency, Green software procurement and Green software policy), mediating variable (software practitioners) and dependent variable (sustainable practice adoption). The limitation of the model is that the model can only support software practitioners to diffuse sustainable practices when they use software applications only, the model cannot support software practitioners to diffuse sustainable practice when they develop and implement software application and services, thus the model is not suitable for software practitioners to adopt in software development process.

Presently this research is in its theoretical phase thus the next phase/future work of this research will involve the validation of a proposed model based on adopting quantitative research by using questionnaire to collect data from software practitioners in selected Malaysian based organizations. The questions in the questionnaire is been developed based on the items of the variables shown in Table 3 of this research paper. Software practitioners in software based organizations are asked questions based on how they adopt sustainable initiatives and strategies in their organizational process. The item in Table 3 will be measured based on a 5 point likert scale ranging from not implemented as “1” and fully implemented as “5”. Structural Equation Modelling (SEM) will be used to analyze the questionnaire data. The model reliability will be tested using Cronbach's alpha and model validity will be tested using regression analyses.

Acknowledgment

This research is financially supported by Universiti Malaysia Pahang; Doctorate Scholarship Scheme (DSS) and Universiti Malaysia Pahang Research Grant No RDU1603118.

References

[1] S. L. Hart, “Beyond Greening: Strategies for a Sustainable World”, Harvard Business Review, USA, 1997.

[2] M. S. Daphne and B. Anol, "Green IT Adoption and Sustainable Value Creation", Twentieth Americas Conference on Information Systems, Savannah, pp. 1-16, 2014.

[3] S. Nils-Holger, E. Koray, M. K. Lutz, and Z. Rüdiger, “Towards a Procedural Model for Sustainable Information Systems Management”, Proceedings of the 42nd Hawaii International Conference on System Sciences, pp. 1-9, 2009.

[4] C. G. Navneet, and G. M. David "Understanding the Enablers and Inhibitors of Decision to Implement Green Information Systems: A Theoretical Triangulation Approach", Twentieth Americas Conference on Information Systems, Savannah, pp. 1-10, 2014.

[5] M. Alemayehu, A. Ahmad, and C. Vanessa, “Green IT beliefs and pro-environmental IT practices among IT professionals”, Information Technology & People, vol. 27, 2014, pp. 129 – 154.

[6] A. J. Bokolo, and C. P. Noraini, “A Framework for Adoption and Implementation of Green IT/IS Practice in IT Governance”, Proceedings of Third International Conference on Green Computing, Technology and Innovation (ICGCTI2015), Serdang Malaysia, pp. 38-48, 2015.

[7] A. J. Bokolo, and A. M. Mazlina, “Development of a Green ICT Model for Sustainable Enterprise Strategy”, Journal of Soft Computing and Decision Support Systems. vol. 3, 2016, pp. 1-12.

[8] R. T. Watson, M. Boudreau, A. Chen, and M. Huber, “Green IS: Building Sustainable Business Practices”, Information Systems: A Global Text, vol. 1, 2008, pp. 1-17.

[9] A. Sulaiman, N. M., Muzamil, and D. Shahin, “Impact of adoption of Green IT practices on organizational performance”, Article in Quality and Quantity Springer, vol. 1, 2015, pp. 1-20.

[10] M. Alemayehu and C. Vanessa, “Green IT Readiness: A Framework and Preliminary Proof of Concept”, Australasian Journal of Information Systems, vol.16, 2009, pp. 5-23.

[11] A. J. Bokolo, and C. P. Noraini, “A Case Based Reasoning Architecture and Component Based Model for Green IS Implementation and Diffusion in Organisation”, International Journal of Digital Information and Wireless Communications, vol. 6, 2016, pp. 97-111.

[12] L. Katrina, K. O. Anni-Kaisa, and R. Paavo, “Supply risks as drivers of green supply management adoption”, Journal of Cleaner Production, vol. 1, 2014, pp. 1-9.

[13] P. Birgit, “Infusing Green: Requirements Engineering for Green In and Through Software Systems”, Third International Workshop on Requirements Engineering for Sustainable Systems, pp. 44-53, 2015.

[14] B. Paolo, G. Qing, and L. Patricia, “A systematic literature review on green software metrics”, Brazil, 2013, pp. 1-25.

[15] A. J. Bokolo, and A. M. Mazlina, “Green IS for Sustainable Decision Making in Software Management”, Journal of Soft Computing and Decision Support Systems, Vol. 3, 2016, pp. 20-34.

[16] N. Stefan, D. Markus, K. Eva, and J. Timo, “The GREENSOFT Model: A reference model for green and sustainable software and its engineering”, Sustainable Computing: Informatics and Systems, vol. 1, 2011, pp. 294–304.

[17] D. Markus, D. Jakob, K. Eva, and N. Stefan, “Green Software Engineering with Agile Methods”, GREENS 2013, CA, pp. 78-85, 2013.

[18] Albert, H. H. “A Model for Environmentally Sustainable Information Systems Development”, Journal of Computer Information Systems, vol. 49, 2009, pp. 114-121.

[19] M. Mohankumar, and M. K. Anand, “A Green IT Star Model Approach for Software Development Life Cycle. International Journal of Advanced Technology in Engineering and Science, vol. 3, 2015, pp. 548-559.

[20] L. Patricia, K. Rick, M. Niklaus, M. Maurizio, A. M. Hausi, and P. Frances, “Exploring Initial Challenges for Green Software Engineering ACM SIGSOFT Software Engineering Notes, vol. 38, 2013, pp. 31-33.

[21] T. Juha, “Good, Bad, and Beautiful Software-In Search of Green Software Quality Factors”, Green ICT: Trends and Challenges, vol. 12, 2011, pp. 22-27.

[22] M. Alemayehu, C. Vanessa, and S. Pittayachawan, “The Green IT Readiness (G-Readiness) of Organizations: An Exploratory Analysis of a Construct and Instrument”, Communications of the Association for Information Systems, vol. 29, 2011, pp. 67-96.

[23] M. Alemayehu, “The Reach and Richness of Green IT: A Principal Component Analysis”, 20th Australasian Conference on Information Systems, Melbourne, pp. 754-764, 2009.

[24] M. Alemayehu, “Organizational Motivations for Green IT: Exploring Green IT Matrix and Motivation Models”, Pacific Asia Conference on Information Systems, PACIS 2009 Proceedings, 2009, pp. 1-13.

[25] K. Stan, C. Vanessa, D. Hepu, M. Alemayehu, and P. Siddhi, “Antecedents to Greening Data Centers: A Conceptual Framework and Exploratory Case Study”, The 21st Australasian Conference on Information Systems Brisbane, pp. 1-10, 2010.

[26] M. Alemayehu, “GITAM: A Model for the Adoption of Green IT”, 19th Australasian Conference on Information Systems, pp. 658-668, 2008.

[27] B. Jose, N. P. Maria, and T. Javier, “Information Technology-Enabled Innovativeness and Green Capabilities”, Journal of Computer Information Systems, vol. 51, 2010, pp. 87-96.

[28] L. C. Chun, and E. W. T. Ngai, "A Research Agenda on Managerial Intention to Green IT Adoption: From Norm Activation Perspective", PACIS 2014 Proceedings. Paper 242, pp. 1-12, 2014.

[29] L. Fabian, “Green IT and Green IS: Definition of Constructs and Overview of Current Practices”, Proceedings of the Nineteenth Americas Conference on Information Systems, Chicago, Illinois, August 15-17, 2013, pp. 1-13.

[30] A. Molla, and V. Cooper, "Greening Data Centers: The Motivation, Expectancy and Ability Drivers", Proceedings of the European Conference on Information Systems (ECIS) 2014, Tel Aviv, Israel, pp. 1-14, 2014.

[31] M. Alemayehu and A. Ahmad “Organizational Green Motivations for Information Technology: Empirical Study”, Journal of Computer Information Systems, vol. 52, pp. 92-102, 2012.

[32] M. Alemayehu, P. Siddhi and C. Brian, "Green IT Diffusion: An International Comparison", Green IT Working Paper Series, 2009, vol. 1, pp. 1-15.

[33] World Commission on Environment and Development (WCED). Oxford New York: Oxford University Press, 1987.

[34] C. Jordi, E. Steve, H. Jennifer, M. Jose-Norberto, L. Lysanne, and L. Sotirios, ”Integrating Sustainability in Decision-Making Processes: A Modelling Strategy”, 31st International Conference on Software Engineering, ICSE-Companion, 1-4, 2009.