

Assignment number: Assignment 02
Unique number: 816583
Student number: 55160735
Last name: Mtombeni
Initials: P.M
Name of Course: Software Engineering
Due date: 31 July May 2018
Lecture Name: Prof E Mnkandla

Abstract

Ethics of software engineering may be approached from 3 directions. 1st, it may explain the software engineer's activity making choices that are practical which may have an impact on some people in ways that are significant. 2nd, it may be utilized in describing collection of ethical imperatives, principles, or guidelines that guiding legislative action, and 3rd, it may be utilized in naming a discipline which studies the relationship among another 2 senses of ethics. Ethics software engineering clearly is both a body and an activity of principles. The software engineering ethics discipline which studies such activity and formalizes such principles, nevertheless, is in its rise. Software Engineering Ethical Activity

To ignore confusion, "ethics", as explained here, is addressing each action that intentional which may positively or negatively affect the values and lives of others. Software engineering is conceiving itself firstly as technical discipline which is developing software. There is adversity of names, like information systems analyst, for people that are engaging in professional software development. Besides the title utilized, the concentration of activity of software engineering is firstly on the technical adequacy of developed products.

Table of Content

1. Introduction.....	4
2. Codes of ethics for software engineers	5
3. Goodness of actions in codes of ethics.....	10
4. Values of human dignity in codes of ethics.....	12
5. Consequences of past unethical behaviour in software development.....	14
6. Conclusion.....	19
7. References.....	20

1. Introduction

This assignment focusses on characteristics of codes of ethics for software engineers. The important aspects or concept of codes of ethics are discussed in this assignment are for example IEEE/ACM code of ethics, Goodness of actions, Values of human dignity in codes of ethics. The IEEE/ACM code of ethics is one code of ethics with eight principles. This paper looks at an entire code of ethics, i.e. the ACM/IEEE and compares it with ACS which is another similar code of ethics for software engineering or the other code that IT professionals use develop software. This paper will tackle consequences of past unethical behaviour in software development. This paper will discuss of any two most common codes of ethics for software engineers and discuss in detail how each chosen code of ethics entails the following practices of ethical behaviour for software engineers, Goodness of actions and consideration of the absolute values of human dignity. The paper will demonstrate consequences of past unethical behaviour in software development.

2. Codes of ethics for software engineers

Bott et al., (2001) states that other disagreements which are arising in computer ethics are depending on tackling ethics from 2 contradictory directions; 1 a rights-duty based ethics and the other virtue depending on ethics. Some people's recent experience with the Software Engineering Code of Ethics development and Professional Practice is leading other in believing that when the differences among such ethical positions is made clear then other problems in reasoning about ethics of computer can decrease.

Chuck et al., (2005) the Computer Society of ACM and the IEEE- have followed the Software Engineering Code of Ethics and Professional Practice recently as the level of quality for practicing and teaching of software engineering. The task force of ACM/IEEE-CS on Software Engineering Ethics and Professional Practices created such code for a sub-specialization among the constituencies of both professional societies. The task force composition is multinational in both membership and in citizenship and in professional computing companies.

Card (2010) suggest that there were 2 ethical approaches of the task force members to this project, yet there was consensus achievement among them. Many ethical discussions regarding the Code relate to virtue based ethics versus relative position of obligations/rights based ethics. It is reflected, for e.g., in values that are different which are placed on privacy. Such difference among virtue and rights based ethics as the first assumption reflected in views that are differing on the acceptable and actual roles of governments in the process of software development and reflected in different obligations or values relating to the roles of individuals and government when protecting data integrity.

The learned lessons from creating the Software Engineering Code of Ethics (www.cs.etsu.edu/seeri) and relating such approaches to ethics offer indications on how to enhance the education of ethics for computer professionals. The paper's observations were done when author's tenure as chair of the task force creating the Code of Ethics. The generalizations in paper, such as each generalization is subject to exceptions. The Code development was an iterative process. The task force reviewed number current codes of ethics from engineering societies, computing societies, and some professions like accounting.

In addition, we considered the functions and goals of such codes and then, through e-mail, chose out various essentials to be put in the code of software engineering. Additionally, it contributed new essentials relating to our software engineering knowledge. Immediately there was an observable difference among Northern European and North American concerns and contributions (Dromey, 2014).

As early drafts were circulated to co-workers for feedback, in addition the co-workers feedback appeared to consider difference in their approach to ethics; 1 was an obligations/rights based ethics and the other was a virtue based ethics. The Code of Ethics draft which resulted was reflecting these 2 various approaches to ethics. The task force members were not mentioning these 2 schools of ethics, but their feedback was easily categorized like it belongs to 1 or another school of ethics. What is the difference among right/obligation ethics and virtue ethics?

There are 2 approaches to morality previously based on the views of Aristotle and Kant, the past representing virtue ethics and the subsequently representing obligation/rights ethics. The ethics of virtue is built on the concept of virtue interpreted relating to specific natural dispositions which people are having regarding human wellbeing contradictory to harm. Ethics of Obligation or rights is denying these natural dispositions, yet highlights the significance of obligation and duty as they are usually expressed in conduct rules found out by the utilization of practical reason.

The view of obligations-right is demanding precise rules for specific behaviors. Commitment to such behaviors can be measured clearly. It is the Kantian moral law; principles of moral were true and were categorical and independent of local circumstance or personal. On the other hand, for Aristotle, the moral rules purpose was development of a moral character or good will, and promoting individual moral virtues.

Put in extra general terms, the obligations/rights ethicist begins with rules stating obligations regarding how 1 must behave and rights regarding how people must be treated, while virtue ethicist begins with the character of human and its ethical dispositions. Virtue ethics cannot lie because of a set of rules that are well defined yet it lies in 1's character; you must recognize the right action and then decide to do it.

Both approaches to ethics were present in the responses of task force to Code drafts. The Code approach indicates 1 way to see the difference among the two approaches and to accommodate each in the agenda of computer ethics.

How these 2 approaches to ethics were reflected at the beginning of development and feedback to the Code? There is number code of ethics purposes. Many principles which were suggested for the code utilized imperative language. For e.g., Ensuring they fully understand the software specifications which they are working on. Ensuring they are qualified, by experience and education, for every project which they are working on. Ensuring proper objectives and goals for every project which they are working on. There are 2 issues with such imperatives for people's obligation. IEEE-CS/ACM decided joint Task Force on Professional Practices and Software Engineering Ethics this was Approved jointly by the IEEE-CS and the ACM as the standard for practicing and teaching software engineering. The small version of code is summarizing aspirations at abstraction which is a high level. The clauses which are involved in a full version provide details and examples of how such aspirations are changing the manner software engineers act as professionals of software engineering. With absence of the aspirations, details may become tedious and legalistic and; with absence of the details, the aspirations may be sounding high yet empty; jointly, the details and the aspirations form a code which is cohesive. Software engineers must be committed in making the maintenance, testing, development, design, specification and analysis of software a respected and beneficial profession.

Bringing to an agreement with the commitment of software engineers to the welfare, health, and safety and of the public, software engineers must stick to the principles. Software engineers must act in a manner which is in the best interests of their employer and client, by being consistent with public interest. Being honest regarding every any limitation of their education and experience.

Keeping private every confidential information consistent with the law and the public interest . Not knowingly using software which is retained and obtained either unethically or illegally .
PUBLIC - Software engineers must be acting consistently with the interest of public.
EMPLOYER AND CLIENT – they must act in a way which is in the best interests of their employer and client .
PRODUCT – the must assure that their related modifications and products are meeting the highest standards possible for professional.
JUDGMENT – they must maintain independence and integrity in their professional judgment.
MANAGEMENT - Software engineering leaders and managers must promote and subscribe to ethical approach to the maintenance and management of software development.
PROFESSION - Software engineers must move forwards in a purposeful way the reputation and integrity of the profession consistent with the public interest.
COLLEAGUES – they must be supportive and fair to their colleagues.
SELF – they must take part in lifelong learning about the their profession practice and promoting ethical approach to practice of the profession.

Such imperatives are not acceptable to people's obligations reason being that they were explaining the meaning of potential obligations that can be legally binding the software engineer. Such formulations were declined reason being there was no way to evaluate if the software engineer satisfied them. Other obligations that human beings were concerned regarding laws being approved which expressed such imperatives. In addition imperative people had problem with words such as "appropriate" and "proper" reason being that they had no meaning that is precise which can be clearly expressed in a measurable obligation or right .

The virtue people did interpret such clauses to say that one complied with imperative when an individual tried his best meeting such standards, acted from a moral disposition . For virtue people, terms such as "appropriate" or "proper" were having a standard, called the standard of being completed from ethical disposition. To cater the obligation human beings words such as "assure" were switched with "aspire" or "strive".

"3.07. Make great efforts to completely understand specifications for software on which software engineers work."

Such substitution is not completely satisfactory to the virtue human beings which characterized such imperatives as "weak" and "wimpy" . The issue is how to keep both groups of software engineers happy. 1 of the important objectives in developing code of ethics is establishing profession which is a consensus . Both groups required to be pleased having profession which is represented by the Code. The approach which has taken in Code of Ethics in resolving such tension was examining distinction among such kinds of ethical theory among professions. Such distinction has been added in early preamble draft of the Code. The distinction among such theories was therefore added in terms of aspirations and rules among the Code.

In preamble to code of early draft 3 professional ethics levels was explained.

"Every Principle of this Code is addressing 3 ethical obligation levels which are owed by software engineers professional in every of such relationships. 1st level was identifying a group of ethical values that they were sharing each and every human being by their humanity virtue . 2nd level is obliging professionals to a higher sequence of care for individuals that can impacted by their task . 3rd and a level which deeper contains several obligations that are directly deriving from elements unique to software engineering professional practice . The Clauses of every Principle are demonstrations of the different obligation levels of added in that relationship(Bott, et al 2001).

ACS Code of Ethics: As a member of ACS one should advance and uphold the effectiveness, dignity and honour of being a professional. It means, in addition to acting within the law and being a good citizen , his commitment to the following values of society :1. Professionalism, 2. Professional Development, 3. Competence, 4. Honesty, 5. The Enhancement of Quality of Life, 6. The Primacy of the public Interest

Such Code of Ethics is applying to each and every member of ACS besides their specific area of expertise or role in the industry of ICT. Professionalism: One must improve society's integrity and respect of its members . Professional Development: individual must improve his own professional development, and that of his staff and colleagues.

Competence: one must work diligently and competently for his stakeholders. Honesty: one must be honest in his representation of products, services, knowledge, and skills. The

Enhancement of Quality of Life: one must make great efforts to achieve to improve the quality of life of those impacted by his work. The Primacy of the public Interest: one must put the public interests public above those of sectional,

personal, or business interests. The Code of Ethics must be adhered to in conjunction with the Code of Professional Conduct.

Professionalism: The industry of ICT is characterised by rapid change and relatively new . This has not had chance to progress over number of years and acquiring its own legislation and standards. In addition it may offer assistance to take action which is appropriate, involving whistle-blowing, when one is discovering a member of ACS engaging in behaviour which is unethical. Members must take note that the ACS may assist them in resolving ethical dilemmas. Discrimination is a behaviour which is unprofessional, like any form of harassment. Each and every human being has the right to be treated with respect and dignity . It is crucial that society members are maintaining professional standards which enhance and improve the image of industry , particularly, in the workplace. The ACS is attempts to improve confidence of the public in the industry of ICT.

Professional Development: Taking action to assure that member's hard-won experience and knowledge are passed on in a manner that the recipients are only not improving the effectiveness their own in their current work, but in addition are becoming eager to take on additional responsibilities and advance their capabilities. Another members can expect one to provide advice and special skills ; and in order to that, one should keep his knowledge up-to-date. One must encourage his colleagues and staff to do the same. One must keep himself informed of new standards, practices and technologies, as they are relevant to his work.

Competence: One may not potentially be knowledgeable on each and every ICT facet yet he must be able to realize if he requires extra information and expertise. It is distinct to accept a task of which the successful accomplishment needs expertise additional to his. One must only accept this work if he believes he is competent to perform it. One must be aware of his own limitations at all the time and not knowingly indicate that he has competence that he does not possess. One should not delay obtaining more expertise from individuals who are appropriately qualified where advisable.

Honesty: One must not breach certain trust of stakeholders or trust of the public in the profession . Circumstances can definitely arise during the process of member's professional career where it can appear to be favourable for him to be fraudulent in some way. This kind of behaviour is not acceptable professional conduct.

Observance of greatest integrity and honesty should underlie each and every all member's professional actions and decisions.

The Primacy of the Public Interest: The interest of public is taken to involve issues of environment, public health, and safety. In his work, member must safeguard the interests of his immediate stakeholders, provided that such interests are not conflicting with the loyalty and duty he owes to the public. Each conflict must be resolved in favour of interest of the public The public interest is taking precedence over sectional, personal, and private interests

3. Goodness of actions in codes of ethics

Dromey (2014) suggest that professional ethics are set of code of conduct, rules and regulations that are governing how a profession is dealing with each other and others outside of the profession. We have different from moral code and different from legal code. The reason why we must have a professional code of ethics is that professional code of ethics serves several functions: provides guidelines in "grey areas", it expresses ideals to aspire to, it codifies members rights, protecting the group's interest, promotes and defines the professionalism of the group.

The idea is to clearly set out international standard for professional practice in IT. The components of the standards are: Maintenance of competency-Practitioners should be able to give evidence of their maintenance of competence. Best practice and proven methodologies -practitioners must be familiar with current best practice and relevant proven methodologies Professional experience: Additional to demonstrated mastery of the body of knowledge a minimum of the equivalent of 2-year supervised experience is recommended before the practitioner operates unsupervised(Gotterbarn,2015).

Training and education -the minimum level of mastery of the body of knowledge must be at the baccalaureate level. Established body of knowledge. The Clauses under every Principle are consisting of 3 various kinds of statement that corresponds to every level. Level 1: Aspire to be human, Statements of aspiration offer objectives and vision, intend to direct behavior of professional. Such directives needs critical ethical judgement. Level 2: Expect to be professional, Statements of expectation is expressing obligations of each and every professional attitudes and professional. Additionally they are not describing the certain behavior details yet they without doubt are indicating responsibilities of professionals in computing. Level 3: Demanding to utilize good practices, demand of statements are asserting extra identified behavioral responsibilities among software engineering that are more relating to present state of art. The statements range is from more common aspirational statement to certain requirements that are measurable."

The Code currently involves imperatives that are at level three expressing obligations that are measurable for rights of their customers and software engineer. In addition the Code is containing statements of level one for virtuous and aspiration objectives for each software engineer. These 2 approaches to ethics are not unsuited, yet this is important in accommodating and recognizing their differences. When it is not completed therefore computer ethics discussions are not resolved. The Code developed gradually following study which is extensive for many engineering and computing codes. Each and every code is taken into consideration trying to inspire and educate the professional group members which is adopting the code. In addition such codes are informing the public regarding the accountabilities which are essential to that specific profession.

Codes are instructing practitioners regarding the standards which the society is expecting them to meet, and what their peers are expecting of each other and striving. Codes are not legislation and they do not intend encouraging litigation, yet codes are offering practical advice regarding problems which matter to professionals and their customers and codes do inform the makers of the policy. To address computer ethics problems in the classroom and for professional require to involve both approaches. The software engineer like a practicing professional is acting from a care of higher level for the virtue ethics(client) and is conforming to standards of development for profession (obligations/right ethics). Both ethics types are required for PROFESSIONAL engineer.

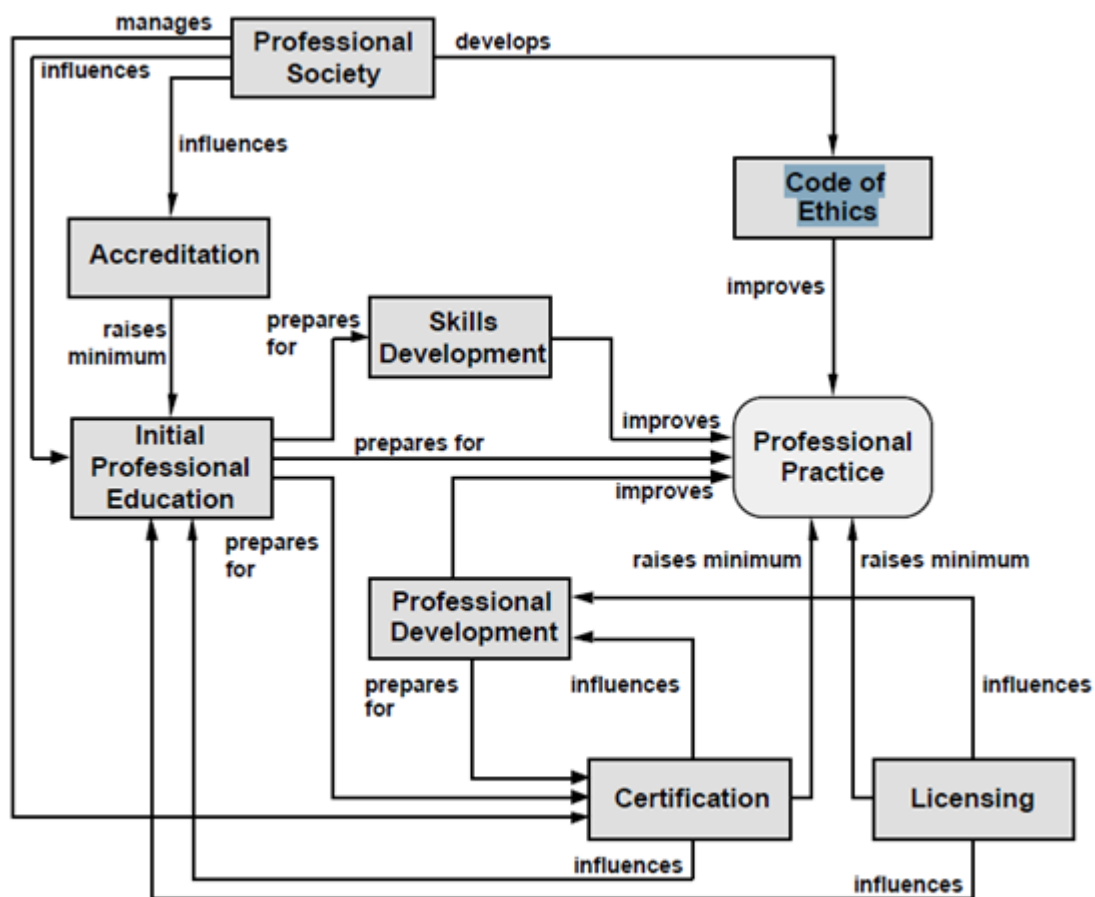
Ethics of software engineering are consisting of 2 significant elements. 1 element named technical ethics, is consisting of making a job which is technically competent at all the software development process phases, another element is the utilization of a group of moral values that is guiding the technical decisions. In the professions, workers technical skill is guiding processes: processes of healing patients, building bridges, of developing software artefacts. The processes performance includes several ethical issues. The choice of physician that dispensed medicine is involving technical judgement regarding its curative powers, and in addition is involving questions regarding medicine's side-effects

Ethics of Obligation or rights is denying these natural dispositions, yet highlights the significance of obligation and duty as they are usually expressed in conduct rules found out by the utilization of practical reason. The view of obligations-right is demanding precise rules for specific behaviours. Commitment to such behaviours can be measured clearly. It is the Kantian moral law; principles of moral were true and were categorical and independent of local circumstance or personal. On the other hand, for Aristotle, the moral rules purpose was development of a moral character or good will, and promoting individual moral virtues.

Ethics of software engineering may be approached from 3 directions. 1st, it may explain the software engineers activity making choices that are practical which may have an impact on some people in ways that are significant. 2nd, it may be utilized in describing collection of ethical imperatives, principles, or guidelines that guiding legislative action, and 3rd, it may be utilized in naming a discipline which studies the relationship among another 2 senses of ethics. Ethics software engineering clearly is both a body and an activity of principles. The software engineering ethics discipline which studies such activity and formalizes such principles, nevertheless, is in its rise. Software Engineering Ethical Activity

To ignore confusion, "ethics", as explained here, is addressing each action that intentional which may positively or negatively affect the values and lives of others. Software engineering is conceiving itself firstly as technical discipline which is developing software. There are adversity of names, like information systems analyst, for people that are engaging in professional software development. Besides the title utilized, the concentration of activity of software engineering is firstly on the technical adequacy of developed products.

The diagram below indicates the goodness of actions in codes of ethics and interactions among components of a profession



Examples of Interactions among Components of a Profession

4. Values of human dignity in codes of ethics

Even though there exist technical standards, there exist many solutions obtainable for most problems of software engineering. The solution option to utilize is built on professional values. Therefore, professional values are having an impact which is direct in a way people are developing applications and quality of the artefacts (Johnson ,2004).

Ethics of software engineering are consisting of 2 significant elements. 1 element named technical ethics, is consisting of making a job which is technically competent at all the software development process phases, another element is the utilization of a group of moral values that is guiding the technical decisions. In the professions, workers technical skill is guiding processes: processes of healing patients, building bridges, of developing software artefacts.

The processes performance includes several ethical issues. The choice of physician that dispensed medicine is involving technical judgement regarding its curative powers, and in addition is involving questions regarding medicine's side-effects. Is this likely to cause another issue? Is this addictive? Is this too expensive for the patient to buy?

Gotterbarn et al., (2009) suggests that technical decisions that are taken in software engineering are comparable in certain respects to this. The life cycle choice to utilize can be leading to products that are radically different, while the both versions of the product can meet the expectations of the requirements.

Generally, each professional ethics (medical ethics, engineering ethics, and legal ethics) are only well known by idea to which they are applying moral rules. Of course, there are differences in professional ethics. The ideas are bringing out different ethical issues. Since contexts that are different are raising ethical concerns that are different, the sequence in which moral rules are put to practical use are varying for every application domain. Even among the process of software engineering, moral rules are offered various importance at various development life cycle stages. Take into consideration various ways informed consent is dealt with.

Throughout the phase of requirements of life-critical software, understanding agreement (consent) that is informed is a rule which important. Throughout testing, principles regarding not cheating and not deceiving are very critical yet consent that is informed is not an important concern.

Different rules and values are applying to different software development life cycle phases of and application of such values in decisions making include no law violation. In the phase of testing, when funds are worn out prior testing is completed satisfactorily and there is no further funds possibility, there are many options. Regardless of which option is chosen it should be controlled by moral rules like "act professionally", "don't deceive", and "keep promises."

Based on software type which being tested, rules like "don't cause pain" and "don't kill" might also come into play. Explains the values of the company procedures that the personnel can follow covers potential ethical issues procedure for handling issues "It is social order that needs constant enhancement, this should be animated by love, based on truth and built on

justice; in freedom it must develop on daily basis going on a more humane balance.” The social nature of the human person and human dignity are the inspiration and foundation for society’s moral vision that can be obtained when people live by the following values and principles: Values

1. Solidarity: we work in solidarity with human beings that are marginal and poor, by that means obtaining the human development, fruits of peace and justice. Solidarity is binding people jointly in the common vision of finding a world where all humans obtain what is rightly belonging to them as daughters and sons of God.

“Solidarity is highlighting in a specific manner the path that is common of peoples and individuals going on an ever more committed unity, the equality of all in rights and dignity, and the essential social nature of the human person.

2. Respect: people respect customs, structures, culture and religious traditions in so far as they uphold and enhance the human person’s dignity.

“A just society is becoming a reality only if it is depending on respect that goes beyond the range of normal dignity of the human person.”

3. Preferential Option with and for the Oppressed and Poor: in a manner conforming with the Gospel of Jesus, people choose to show those who are oppressed, poor, or marginalized. People are dedicated to act in preventing the dehumanizing poverty which is robbing others of their humanity and dignity. People are guided by Christian Scripture to assist the marginalized be accountable for their own development and an equitable share of gifts of the earth, and in working for freedom of the oppressed.

5. Consequences of past unethical behaviour in software development II.

Dysfunctional behaviors and ethically doubtful actions are making it hard for developers to be trusted by other IT staff and users. 1 of the difficulties encountered by IT professionals globally is that they are frequently not trusted by humans they are meant to be serving (Johnson, 2004). There is number of reasons why IT professionals are not trusted, but the first 1 is that IT professionals are frequently regarded as liars and in accordance with a legitimate right considering the record track of software development.

Principle disagreement in the senior management policies. Your current employer may act in an unethical way and decide to release a safety-critical system without completing the system testing. The outcome connecting many conferences, having number of discussions with professionals and experts, and reading various articles relating to ethics maybe concluded as: ethics are having number definitions, which are reflecting the viewpoints of authors or philosophers and their culture, yet there is a general area among each viewpoint (Kreutz et al.,2015).

The ethics may be defined as "Motivation based on ideas of right and wrong", "Set of principles of right conduct" or "Theory or system of moral values". Wikipedia encyclopedia indicates that: Socrates was 1 of the 1st Greek philosophers to persuade both the common citizen and scholars to turn their awareness from the outside world to the man's condition. Aristotle submitted ethical system that can be termed "selfrealizationism"; when an individual is acting in accordance with their nature and realizing their full potential, they can be content and do good (Laudon, 2005).

Roche (2012) states that individuals are facing ethical issues on daily basis at their life; but the question is how many of us are having the knowledge of how to deal with them? Number of surveys were completed that indicate the entire array of problems that are faced by workers like: deceit, fraud, lying, and stealing. Internationally, in addition ethical values are deficient.

In 300 company surveys that were done globally, more that eighty five percent of senior executives showed that the subsequent problems were between their top ethical concerns: unauthorized payments, sexual harassment, inappropriate gifts, and employee conflicts of interest. 2,000 surveys of major United State corporations were revealing that the subsequent ethical issues which concerned managers: misuse of company assets, plant closings and lay-offs, environmental pollution, abuse of company expense accounts, misuse of proprietary information, discrimination in hiring and promotion, quality control issues, conflicts of interest, employee theft, alcohol abuse and drug. In software development and computer, there are number of issues that are relating to ethical issues (Sommerville, 2004). These problems involve software ownership and intellectual property rights, computer crimes, safety, reliability and security, harmful and affected high ratio of people, impact on work place and society, professional competence, liability and responsibility, confidentiality, privacy, quality as moral issue, social responsibility, and professional responsibilities.

Ethical related issues in software and computer are very costly. A study of FBI (federal bureau of investigation) indicates that in two thousand and six the approximate crimes of computer cost USD sixty-seven point two billion every year. Contrarily , software engineers take part in advanced software development as a key for all mass-destruction weapons and intelligent systems (Watzlawick et al., 1967). The usage that is unethical like weapons are causing 100 of 1000 of victims that are innocent and the destructions that is huge for environment and wealth, that means that software engineers are taking part indirectly in this destructions and crimes. A study of survey completed on ethical problems in education and usage of internet; the sample is chosen from Malaysia's higher education institutions.

The users of Internet (non-academic staffs and academic staffs, students) are representing different ages and qualifications, experiences, genders, cultures, races, and nations, this is making the chosen sample of users to represent the population which we require to study. The output of survey indicates that: Fourteen percent of total users view data related to security as data factor that is harmful. Nineteen percent of total users view data related to politics as data factor that is harmful. Nineteen percent of total users view data related to Anti-Culture as data factor that is harmful. Fourteen two percent of total users view data related commercial and Advertisements announcement as data factor that is harmful. Fifty one percent of total users view Anti-religious as data factor that is harmful (Yongchang , 2010).

Seventy three percent of total users view data related to sexual harassment as data factor that is harmful. In addition, it indicates that seventy five percent of the users receive this harmful data through e-mails. Discussion in this paper is relevant mostly to research projects which is employing human subjects or which includes the information collection which may be leading to individual's identification.

This information which is identifiable may gathered via the human observations or the artifacts examination for example documents or source code. Notice that it is covering most of research of empirical software engineering whether this includes process studies, metrics, or workplace studies.

. In addition, the discussion is applying to the practice of Software Engineering if this includes identifiable information or human subjects, usability testing for e.g., via the main concentration which is on research of ESSE. This research is not applying to another software engineering practice research or areas, like the components research or the application or development of standards. For discussion of issues of ethics that arise in such areas, there exists several excellent books for example ([39], [38], [31], [19]).

In addition, the reader which is interested is directed to the Practice of ACM/IEEE-CS Software Engineering Code of Ethics and Professional [15], [17], and the Professional Conduct and ACM Code of Ethics [1]. There are number of reasons why potential subjects of ESSE, researchers, and corporate sponsors must be worried about with research ethics (Bott et al., 2001).

Production of responsibility in the of difficult software systems .Software systems are characterized strongly by their difficulty: both the delivered product and the production process are realities that are very complex, even more when developers are considering the vast demand of extra heterogeneous, distributed, and networked systems.

Developers may take into consideration 2 various implications of difficulty regarding ethical responsibility. 1st, responsibility can be shared between a large group of human beings, from designer to tester, from requirements engineer to software architect, and above each project manager. Who is going to be accountable for failures in the software that is delivered, for exceeding the budget, for missing scheduled deadlines? What if people lives, jobs, or houses are relying on the software system which working correctly, or easily delivered on time?

Such demands a clear responsibility distribution between various roles in the process of production. 2nd, complexity is making effect more unpredictable, and hence a consequentialist analysis of responsibility is becoming harder. Everyone will be very excited when developers had a perfect manner of predicting the software system behaviour, yet software system is not a world of mechanical of ideal billiard-balls.

Regrettably, computer science has not yet arrived at a level where developers are having, predominantly, formal proofs for the correct functioning of a subsystem or software system. Many times, systems are not proven formally reason being software do not even know how

to; systems are easily tested. When developers are waiting for a demonstration which is perfect the software does not have bugs, therefore no software can be delivered, and it is not what the society is expecting from the software engineers. Testing should be (and can be) rigorous and methodical, and many standards of software engineering is containing clearly defined procedures and guidelines to obtain good software testing. But developers may not anticipate, from testing the mathematical demonstration that system can have no bugs. Software Engineering, same as Medicine, it is not Mathematics.

The fact here is that this unpredictability and imperfection of software is belonging to the software engineering profession nature, as developers have the knowledge of it nowadays, and as is socially and publicly recognized. Fact is, developers find the same unpredictability and imperfection in another fields of engineering. The civil engineer must, and can support measures that are reasonable in avoiding a bridge break-down, yet she cannot mathematically assure it.

When he is adopting these measures that are reasonable, and that are well known in the profession, therefore he cannot be accountable for the collapse, when it happens. Furthermore, developers require a manner of putting boundaries to the effect that the development team must be accountable for. Between the different contemporary schools of thinking of moral, consequence-based ethics, as contrary to rule-based, gives the impression of having a good acceptance between professionals like software engineers. Yet naïve consequentialism is weak in serving as a practical guide in the profession. apart from, the software systems complexity making it extremely difficult knowing in beforehand effect that can derive from professional activities in the software production.

Hence, because of the well-known codes of ethics spirit like the ACM/IEEE's, it is advocated for an extra solid position, that is called "moderate deontologism", that is considering both consequences and rules to assess the goodness of actions, and simultaneously pays an adequate consideration to the absolute values of human dignity. The society's moral progress is influenced highly by the manner individuals reason in particular professional ethics, and in the different fields of ethics. The laws which are governing a society are accountable for the structure it is acquiring in the long term. But this is the work of ethical thinking trying inspire the laws of development. Every 1 of us acknowledges the ethics primacy above law if individuals cry out: *this law is unjust!* (Think of laws in terms of minimum salaries, racial discrimination, and s forth.) Besides the brute force of votes, or of weapons the other only force which may be changing the laws is the reason which is ethical. This is the reason why ethical thinking is essential in everyone's education: when their moral arguments are not strong, they are at the best speaker's mercy. In detail this is essential in the modern professionals education, like software engineers, reason being that the ethical thinking is only not made up of principles which are abstract, in addition this is derived from real circumstances and professional life. When an individua tries to formulate ethical judgments regarding salaries, taxes and rates of interest, he should knowledgeable regarding such notions in the economy field. In a similar manner, judgement regarding moral accountability of the software engineer needs a good profession knowledge, well informed about the real way engineers work and the experience.

Each engineer is 1st of all an ethical agent, a free person. Ethics, which are far from being a group of limits imposed on individual's freedom, is a precise manner individuals become their own masters. Without a solid and specific ethical education, the engineer is becoming just a depersonalized, technical instrument in the hands of other individuals.

This was recognized in many educational institutions and places. In detail, the *Computing Curricula* created by ACM/IEEE, that is taken like an appropriate model for many programs of the university, it putting a sufficiently great emphasis to courses of laws and ethics in Chapter 10 devoted to Professional Practice, and is promoting different strategies to incorporate them into curriculum of computer science.

In this assignment we are not worried about ethical issues that are general in IT, like intrusions, intellectual property of software products, censorship and freedom in the Internet, a privacy of personal data, abuses and frauds committed with the aid against, or of software systems, and so forth. Instead this paper is concentrating on ethical problems which are more directly concern with the accountability of software engineer in the faulty software systems production, and the consequences which are bad that may be obtained from them. Software systems are very powerful systems that may result in harms that are severe to well-being or human lives and if such is occurring people want to know who is going to pay for it, who is accountable. Yet such analysis should not ignore that this is in the nature of Software Engineering dealing with the complex system production, in which the results of actions are specially hard to predict.

Consequences vs Rules is there a clear boundary?

Ethics contemporary schools may be arranged in various ways. A general distinction between them is that of "consequences vs. rules". Ethicists which are in the "rules" camp understand that *good actions is the outcome of adhering the correct behaviour rules*, where normally are thought to be applicable to all and universal; the rules should be followed besides the consequences, bad or good, which may result.

Ethicists who concentrate consequences, contrary, understand general rules are not clearly defined to guide feel and action rather that people should view the consequences of their our actions, and *taking actions which are producing the best consequences or results*

People were engineering things for decades, and every time they came across similar ethical challenges, this was outlined elsewhere. So what is making software engineering remarkably different? Software engineering is a practice with which number of human beings are not unfamiliar and where Problems and issues are difficult to identify in advance. Hence, human beings should trust experts of software engineering more than experts in some fields of engineering. Software engineers is frequently engaging in unethical and unprofessional behaviour not realizing it. Through professional association codes or In courses of ethics, practitioners can be noticed of dilemmas or situations which can finally lead to unethical behaviour, like "Avoid harm to others."

A joint task force of ACM/IEEE-CS has developed a code of ethics to address unethical behaviour issues of (www.nspe.org/Ethics/index.html). Yet frequently the concentration is on headline scenarios and not on (mundane) initially circumstances which abound in their profession. Additionally, ethical training cannot be offered the required priority throughout undergraduate education. Instead of isolated ethical lapse, what usually is making headlines is the outcome of a sequence of ethical lapses that are related. If these scenarios are cascading, there is normally a magnification impact. Unethical behaviour is occurring in software engineering if the professional should be making a choice among values that are competing, like professional vs personal. For e.g., sales manager can be signing a contract to provide software product having been advised, or knowing, that the product can take longer for delivery than promised date. The dilemma of sales manager can be that her employer is pressure in meeting financial target, or there can be consequences of related job.

In a feedback, the project manager develops project schedule that are unrealistic, that may lead development colleagues choose to defer so that they may obtain organizational or political goals. A chain of unprofessional or unethical behaviours can therefore take place which finally leads to bankruptcy, layoffs, lawsuits or even massive penalty payments. But no one in this process can recognize or admit to that their behaviour was unprofessional or unethical.

Categorizing Dilemmas

In shining some light on such types of subtle but nevertheless unprofessional or unethical or situations, each dilemma is given a name. Readers may not agree with the choice of labels, yet we doubt they can disagree that this is inappropriate behaviour. This list cannot be or is not comprehensive. It is discovered that most are significant instances of ethical dilemmas. All those we describe involve common occurrences. We cannot be sure that simply *naming* them will solve anything, but it will help us discuss them. Notice that not every wrong behaviour is unethical. When human beings behave wrongly and do not know better, they may not act unethically. This is certainly unethical, nevertheless, for human beings to make decisions if they know they are lacking the skill required in making professional decisions that sound good. The *ethical behaviour* term is referring to how an organization or an individual assures that each and every stakeholder interactions, decisions, and actions, obey the organization's or individual's professional and moral principles. Such principles must back up all applicable regulations and laws and are the base for the organization's or individual's values and culture. They are explaining right from wrong. In most cases, mismanagement, personal misconduct, unprofessional behaviour, incompetence, or, more frequently, a seemingly insignificant chain of professional lapses or small ethical is bringing about the situations which makes headlines.

For e.g., when a project runs late, the project manager can might be tempted cutting short the requirements definition phase, with the hope of making up for lost time. For them getting the product out the door, software developers are basing their test not on the requirements, instead descriptions of software developer of how their code is going to work. The members of team therefore is delivering the outcome to clients with catastrophic consequences possibility, like a lawsuits, contract cancellation, or an unusable product. Such behaviour is certainly unprofessional and short-sighted at best. Decisions that are wrong are leading to bad results. When an individual who must know better is making wrong decisions, and when personal interests is motivating such decisions, the behaviour is becoming unethical. If learners are enrolling in ethics courses introductory, learners are learning about extreme and clear situations. Such environment is making it relatively simple distinguishing if a behaviour is unethical or is crossing the line. Nevertheless, real life is not so easy, and the next dilemmas are coming from scenarios that are occurring more often.

6. Conclusion

Ethics should be considered in each and every phase of the process, this is the best technique of discovering unethical system properties. Requirements that are toxic must be eliminated or avoided at all the time. In addition Requirements that are deceptive must be eliminated or avoided. Software system development which is making ethical decision is an active research field. Processes that are new are being created to incorporate ethics into .software systems are characterized strongly by their difficulty: both the delivered product and the production process are realities that are very complex, even more when developers are considering the vast demand of extra heterogeneous, distributed, and networked systems

Ethics of software engineering are consisting of 2 significant elements. 1 element named technical ethics, is consisting of making a job which is technically competent at all the software development process phases, another element is the utilization of a group of moral values that is guiding the technical decisions. In the professions, workers technical skill is guiding processes: processes of healing patients, building bridges, of developing software artefacts. The processes performance includes several ethical issues. The choice of physician that dispensed medicine is involving technical judgement regarding its curative powers, and in addition is involving questions regarding medicine's side-effects

Ethics of Obligation or rights is denying these natural dispositions, yet highlights the significance of obligation and duty as they are usually expressed in conduct rules found out by the utilization of practical reason. The view of obligations-right is demanding precise rules for specific behaviours. Commitment to such behaviours can be measured clearly. It is the Kantian moral law; principles of moral were true and were categorical and independent of local circumstance or personal. On the other hand, for Aristotle, the moral rules purpose was development of a moral character or good will, and promoting individual moral virtues.

7. References

Bott, F., Coleman, A., Eaton, J., Diane Rowland, D.(2001). Professional Issues in Software Engineering, 3rd Ed. Taylor & Francis.

Card, D.(2010). "Why Do So Many Reuse Programs Fail?" IEEE Software, pp. 114–115

Chuck Huff, C. ,Dianne Martin , D.(2005). "Computing Consequences: A Framework for Teaching Ethical Computing". Communications of the ACM, 38(12):75-84.

Dromey, R. G.(2014). "A Model for Software Product Quality," IEEE Trans. on Software Engineering, Vol. 21, No. 2, pp. 146–162.

Gotterbarn, D.(2015). "The Moral Responsibility of Software Developers: Three Levels of Professional Software Engineering". The Journal of Information Ethics, 4(1): 54-64,.

Gotterbarn, D., Miller, D., Rogerson, Simon. (2009). "Software Engineering Code of Ethics is Approved". Communications of the ACM, 42(10):102-107.

Johnson, D.(2004). Computer Ethics, 2nd Ed. Prentice Hall.

Kreutz, D. , Ramos, F. M. V. , Verissimo, P. , Esteve, C., Rothenberg, S., Azodolmolky, S.(2015) "Software-Defined Networking: A Comprehensive Survey." To appear in proceedings of the IEEE.

Laudon, K.(2005).. "Ethical Concepts and Information Technology". Communications of the ACM, 38(12):33-39.

Roche, J. (2012) . Adopting DevOps Practices in Quality Assurance, Communications of the ACM.

Sommerville, I.(2004). Software Engineering, 7th Ed. Pearson-Addison Wesley.

Watzlawick, P., Bavelas, P., Jackson, D.(1967). Pragmatics of Human Communication. W. W. Norton & Company.

Yongchang, R.(2010) Fuzzy Decision Analysis of the Software Configuration Management Tools Selection. In ISCA 2010. France. Information Science and Engineering (ISISE): ACM. 295 - 297.