Thoraya Shaheen
16318376

MEASURING SOFTWARE ENGINEERING

CS3012

Introduction

Our task is to see whether the software engineering process can be measured or not in terms of different aspects being the computational platforms available to use for the measurement, the algorithmic approaches available, and the ethics concerns. But first lets look at:

What is Software Engineering?

It is the study of engineering to the design, development and maintenance of software. It's become a huge part of our everyday life in almost every single product that we use and so it has to constantly and become more efficient in order to fit with the needs of everyone from a single user to immense tech enterprises. And so, a software product has to satisfy many categories being scored such as the following:

1. Operational: meaning how good this software hits the budget, efficiency, functionality, dependability, security and safety.
2. Transitional &Maintenance: to be able to change from one platform to the other, and how good a software will work when changing the surrounded environment, portability, adaptability, flexibility.

Stages of Software Engineering

1. It all starts with the client giving specific requests for a certain task and submits it to a service provider organisation.
2. The programmer/team splits the tasks into 3 main categories, user requirements, system requirements and functional requirements.
3. An agreement is achieved and from there gathering data and studying the current system and how to improve/change/add on it.
4. The team analyses whether the software needed to be built will fulfil all the requirements of the user and also the limitations on its system.

5. After that, the software design begins by writing code using a suitable programming language.
6. Various testing levels are performed and that is vital in terms of building any project.
7. Listen to feedback and learn from it.

Why do we need Measurements?

Measuring software is vital because it can facilitate a true engineering approach to Software Development, improve software processes and productivity.

"In seeking to improve software, companies are finding out how much is involved in measuring it. They are also learning that the more integral software measurement is to the company's underlying business strategy, the more likely it is to succeed."

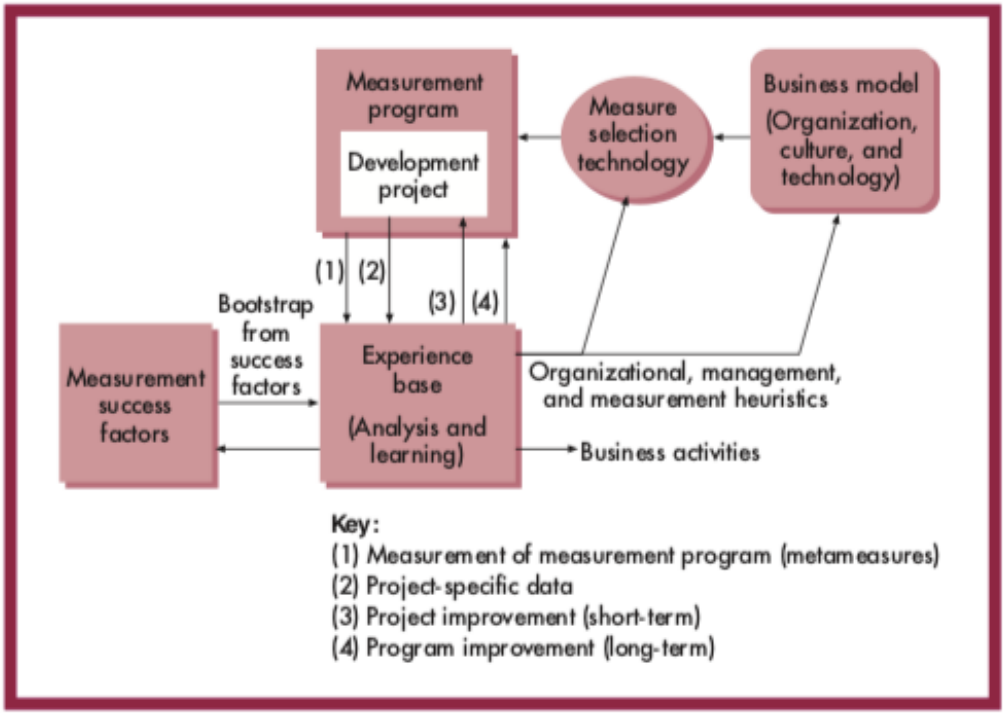

***Figure 1**. The M³P framework. An appropriate measure selection technology links the underlying empirical and numerical measurement models. M³P can be bootstrapped from existing material describing generic measurement program success factors.*

Having a well-defined link between the numerical data and the surrounding development and business contexts by pairing technical, business and organisational issues into a given measurement program context makes a powerful Software that is ready to be produced by the businesses.

Productivity makes a huge factor when measuring Software because it relies on the amount of time spent for completing certain tasks. Therefore, metrics need to be used in order to manage, keep track of productivity and increase it.

These metrics are also vital because they help the managers to assess what influence their decision have on the overall process and therefore be able to communicate better, improve their workflow and report issues easily and quickly enough.

How do we achieve these Measurements?

In order to be able to measure software correctly, there are a few things that need to be taken into perspective such as:

- Code Quality
- Code Complexity
- Algorithms Analysis
- Functional Points Analysis
- Productivity
- Number of developers needed to complete the task
- Code churn (how many LOC are produced, edited, deleted)
- Time taken to fix bugs
- Level of Testing and test cases
- Software Risk
- Software Size
- Meeting the Standards and Regulations
- Customer Satisfaction

And in order to be even more successful at measuring software, these steps need to be done repeatedly every specific periods until the proper outcome is maintained and the goal has been reached.

Computational Platforms used for these Measurements

There are several metrics that are used to measure Software Engineering processes and production environments. Nowadays mostly every software engineering process is associated with the success and efficiency of the businesses.

The Personal Software Process (PSP)

It was based on manual assessment, filling out forms and reports detailing their work procedure, issues and how they tackled the problems. This platform had its pros and cons, pros as in it left space for the developers to just focus on specifying the things they need to analyse and not more but of course this wouldn't have been very practical given that everything had to be done manually meaning it would actually take a much more significant time than it should rather than focusing on implementing and improving just the code itself. So of course, a new strategy had to come to light which brings us to the next one.

Name: Jill Fonson **Program:** Analyze.java

Date	No.	Type	Inject	Remove	Fix time	Fix defect no.	Description
9/2	1	50	Code	Com	1	1	Forgot import
9/3	2	20	Code	Com	1	2	Forgot ;
9/3	3	80	Code	Com	1	3	Void in constructor

FIGURE 1. A sample defect-recording log. In the Personal Software Process (PSP), even compiler (syntax) errors are recorded. Developers typically find this aspect of the PSP to be onerous.

The LEAP Toolkit

LEAP tackles the problems that PSP faced, it automates the data analysis to tackle the data quality issue, this was all still done manually. What it basically does is create a portable repository of personal process data that make it easier to commute from one project to other projects, it allows developers to have total control over their data files, it doesn't focus on the developers name but it just focuses on improving the project by keeping track of their activity, what they added and so on.

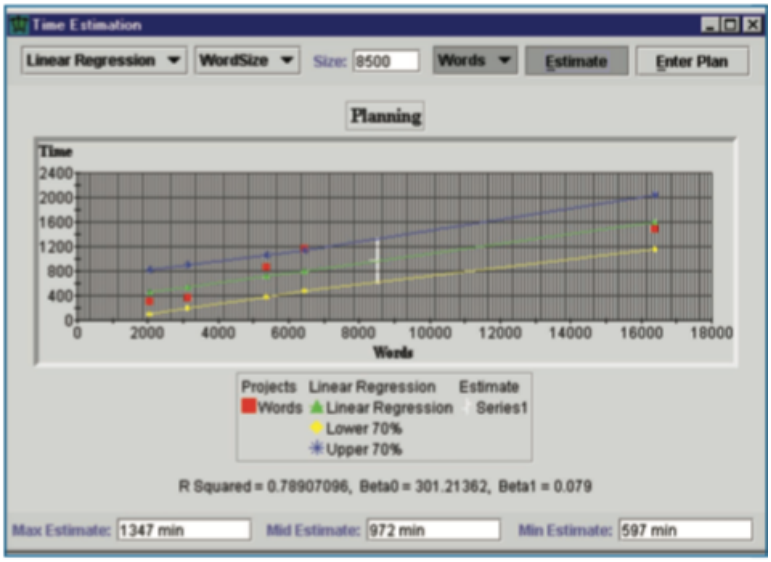

FIGURE 2. The time estimation component in the Leap (*l*ightweight, empirical, antimeasurement dysfunction, and *p*ortable software process measurement) toolkit. Unlike the PSP, no Leap analytics are paper-based.

Hackystat

It was the answer for everything and by everything, I mean involving all sorts of things from collecting the data to analysing, visualising and interpret it all in one for free.

It has 4 major design features:

1. Data collection from both the client and the server.
2. Make sure the process of collection the data isn't disruptive.
3. Collecting data Minute-by-Minute.
4. Involve both personal and group-based development.

The health indicators featuring aggregations of individual developer's behaviour in the software ICU interface showed results for:

- DevTime: how much time is spent on the IDE working on the project's files.
- Commit: how often the developer commits to a repo and how many more LOC were added each time.
- Build: keeping track of the times the program was built and if it was a success or failure.
- Test: keeps track of how the test case tackles different problem and if they passed or not.

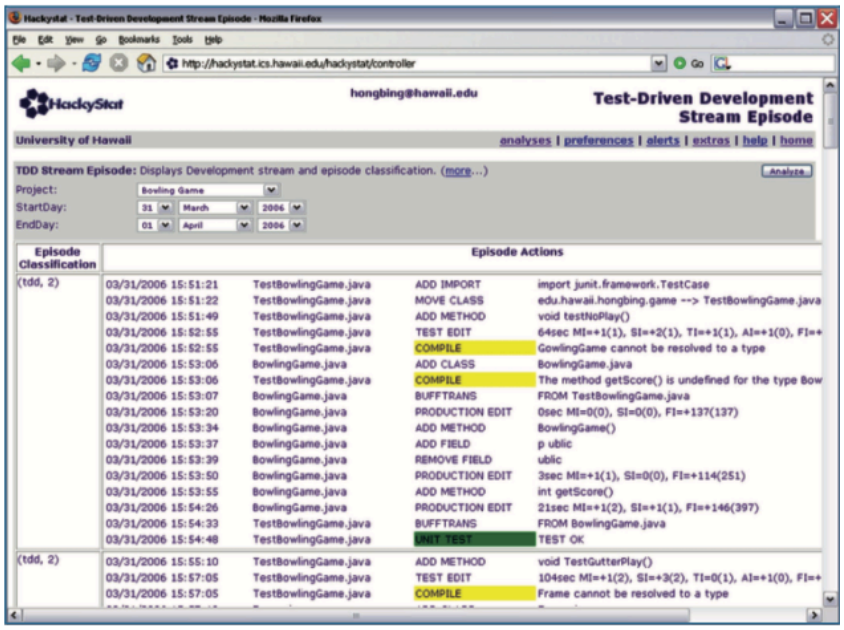

FIGURE 4. The Hackystat-based Zorro system can automatically determine the extent to which developers use test-first design methods. Some developers were uncomfortable with this fine-grained data collection.

Algorithmic Approaches used to analyse Big-Data

1. **K-Means Clustering Algorithm:** it's a simple, unsupervised learning algorithm, it is usually paired with big data sets in order to classify them into greater categories in order for other algorithms to be able to further refine them.

2. **Association Rule Mining Algorithm (Market Basket Analysis):**
It's a learning algorithm that detects any links incorporated between data that occur frequently.

3. **Linear Regression Algorithms:**
It's a widely used method of statistical analysis, it has the ability to tackle many problems, it can detect trends and changes of policies.

4. **Logistic Regression Algorithms:**
It is a classification algorithm. It's used to find the ranging of success or failure of a given event.

5. **C4.5:**
A supervised learning algorithm designed to create decision trees from a classified input. They're used as diagnostic tools in medicine and businesses too.

6. **Support vector machine (SVM):**
SVM can help figure out an underlying separation mechanism between people who will buy a product and those who won't by defining a hyperplane -the line that separates a group.

7. Apriori:
It's a matching algorithm. It is commonly used in transactional databases with a large number of transactions, it runs with a high degree of computational overhead.

8. EM (expectation-maximization):
It is a clustering algorithm used for knowledge discovery. It uses clustering to predict data models that can be used in other statistical analysis methods.

9. AdaBoost:
An algorithm which looks for the best learning algorithm from a list of machines and refines them based on their efficiency. It then distributes the enhanced set of information to the other machines, from which it eventually optimises the ability to learn of involved machines.

Ethics of Data Collection & Analysis

Computer emergence and the vast flow of Information due to the rapid technological upgrades has shaped our lives, not just by providing new tools but also impacted the values and standards of its essence. Consequences of this information age are various and have questioned the ethics and security of almost every industry, every home, every school and every government. Academic literature gives an encompassing term to define the inter-relation amid Information security and ethics and refer it to all activities needed to secure information and systems that support it in order to facilitate its ethical use. According to the National Institute of Standards and Technology (NIST) the definition of Information Security is "Information Security is protecting information and information systems from unauthorised access, use, disclosure, disruption, modification, or destruction in order to provide confidentiality, integrity, and availability." (NIST, 2003, p3). This trio of objectives sometimes is referred to as the "CIA Triad".

Information technology gave birth to computer-based activities which revolutionised the way we interact using new intermediaries and options in which impacted our hierarchy of values (Mesthene, 1968).

The ethical issues involved are many and however according to Richard O.Mason. They can be summarised in an acronym -- PAPA.

Privacy: Information is very crucial today in decision making therefore citizens must ask themselves, What information they should share about themselves and what information should be kept? Should there be condition for disclosure and what are they? The growth of technology and its growing capacity of storage and ability of surveillance is a major issue in terms of user privacy.

Accuracy: This involves the accuracy of the information available. Questions asked like Who is responsible for the authentication of the data? And if errors are found, are they accountable? to what extent?

Property: Questions concerning intellectual property rights are very debatable in the new society. Who owns the means of information transition? Who has the right to copy and reproduce certain information and in what capacity?

Accessibility: basically, asks question regarding the literacy in our modern society. Who has privilege to obtain what information and in what capacity or under which conditions? these questions are also interrelated with property rights.

How ethics can enhance organisational privacy

According to a study done by Culnan, M. J., & Williams, C. C. (2009), conducted on 2 US companies, some practical steps can be implemented to improve organisational security and privacy of clients. The first step is to create a culture of privacy initiated by the CEO or very high levels of management. The CEO will influence the lower rank employees to see that privacy is a high value of the company and therefore will invest in ethical programs and projects and policies that increase moral responsibilities.

Second step is to conduct a Governance Process for Privacy. Last step is to Avoid decoupling, in other words companies must stop using personal information of clients as it is a part of the organisation's complex process but rather think as if it was their personal information that is being used.

References

- Establishing Software Measurement Programs – R.J Offen, R. Jeffery
- Searching under the Streetlight for Useful Software Analytics – Philip M. Johnson, University of Hawaii at Manoa
- Mason, Richard O. "Four ethical issues of the information age." Computer Ethics. Routledge, 2017. 41-48.
- Nemati, H. (Ed.). (2007). Information security and ethics: concepts, methodologies, tools, and applications: concepts, methodologies, tools, and applications. IGI Global.
- Culnan, M. J., & Williams, C. C. (2009). How ethics can enhance organizational privacy: lessons from the choicepoint and TJX data breaches. Mis Quarterly, 673-687.
- Wack, J., Cutler, K., & Pole, J. (2002). Guidelines on firewalls and firewall policy: Recommendations of the National Institute of Standards and Technology. National Institute of Standards and Technology, NIST Special Publication 800-41. 2002. http://csrc.nist.gov/publications/nistpubs/800-41/sp800-41.pdf
- Mesthene, E. (1968). How technology will shape the future. Science, 135-143
- https://economictimes.indiatimes.com/definition/software-engineering