









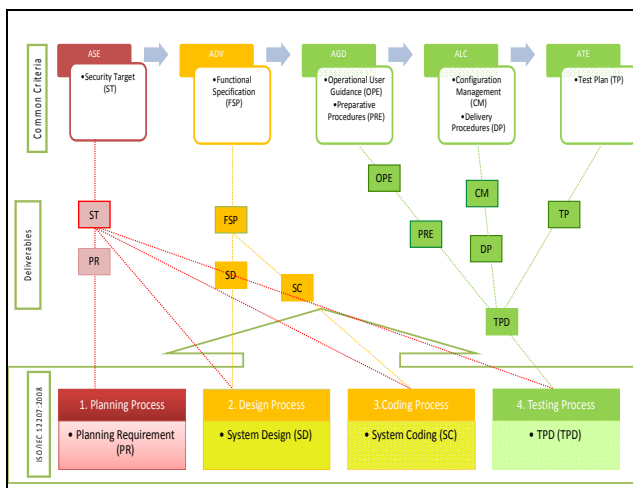
b) The framework of Common Criteria and commonly practices industry of software engineering (Agile Software Development)

The consolidation between process and deliverables between Common Criteria and Agile Software Development has being shown in the catalogue later. The Agile Software Development is found as the most acceptance software process in the industrial practices (Andrew Begel and Nachiappan Nagappan; 2007). The election of being this paper as the primary reference had been decided during the investigation on the software engineering practices in the Chapter 4.

The Agile Software Development consists of several processes which are started by planning, design, coding and test as in the table III. This project only emphasizes the process and deliverables taken in the Agile Software Development and the Scrum model that being used by Agile Software Development is not being concentrated.

TABLE III: THE PROCESS AND DELIVERABLES OF ASD

| Agile Software Development | Deliverables   |
|----------------------------|--|
| 1. Planning Process        | User stories value<br>Acceptance Test Criteria<br>Iteration Plan |
| 2. Design Process          | Simple Design CRC cards<br>Spike Solutions prototypes            |
| 3. Coding Process          | Pair Programming   |
| 4. Testing Process         | Unit Test<br>Continuous Integration<br>Acceptance Testing        |



**XI. EVALUATION & ANALYSIS**

This section presents finding of evaluation that was conducted on the framework. The evaluation is using the questionnaires that were distributed to the respondents. The objective of the evaluation on the framework is to analyze the feasibility and understanding of the framework that was created.

a) *Perceived ease of adoption among respondents*

The figure 7 shows that the certifiers, evaluators and software developers are positive about the ease of adoption of the framework into the environment of Common Criteria evaluation and certification. The higher percentage goes to

the certifiers with 88% followed by the software developers with 78% and the evaluators with 60%. Although of the percentage of evaluators are less than others, it still shows a positive percentage because it more than half percents from the sample of evaluators population. Therefore, it can be concluded that the framework is perceived ease of adoption in Common Criteria evaluation and certification among the sample population.

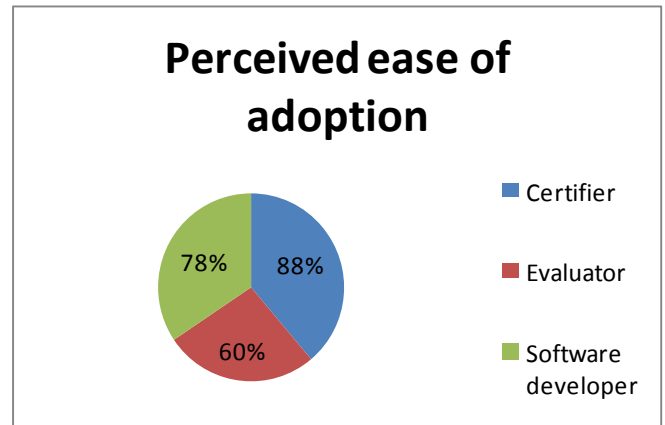


Fig 7: Perceived ease of adoption among respondents

b) *Perceived utility from framework*

The figure 8 shows the productivity level is among the highest and consistent from the respondents. The productivity of evaluation and certification of Common Criteria is perceived among the respondents from the framework. Most of the respondents also agree that less time required during the Common Criteria evaluation and certification when using the framework. The last top three utilities that could effect from the framework is the quality. Most of the respondents perceive that the quality of evaluation and certification of Common Criteria would be enhanced from the framework.

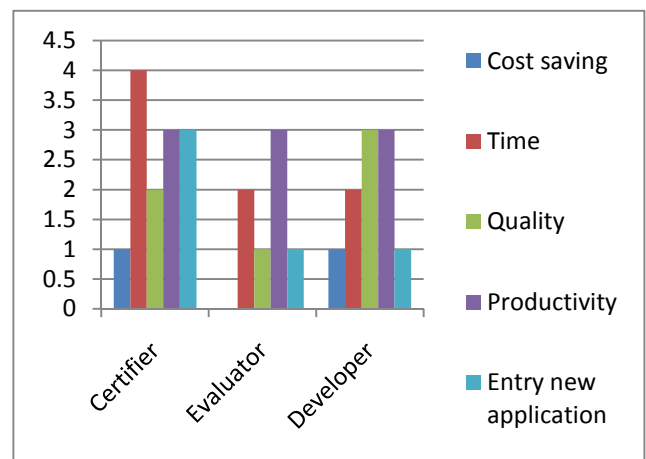
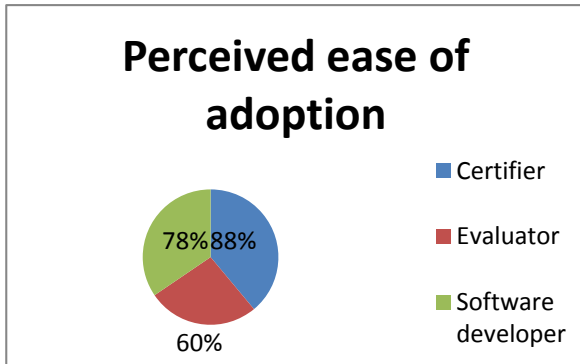


Fig. 8: Perceived utility from the framework

c) *Perceived ease of adoption among respondents*

The figure 9 shows that the certifiers, evaluators and software developers are positive about the ease of adoption of the framework into the environment of Common Criteria evaluation and certification. The higher percentage goes to the certifiers with 88% followed by the software developers with 78% and the evaluators with 60%. Although of the

percentage of evaluators are less than others, it still shows a positive percentage because it more than half percents from the sample of evaluators population. Therefore, it can be concluded that the framework is perceived ease of adoption in Common Criteria evaluation and certification among the sample population.



9: Perceived ease of adoption among respondents

**XII. CONCLUSION**

The integrated framework will facilitate the software developers in producing software complete with security assurance because it is integration between software engineering and Common Criteria. When the integrated framework facilitate the software developers in developing a software complete with security assurance, it will increase the competency among software developers in developing more secure software and encourage more software product to be certified by Common Criteria. The integrated framework will simplify the process of evaluation and certification of security functionality of the product because the process and deliverables from software engineering will be mapped to the process and deliverables from Common Criteria. When the software developers aware about the existence of Common Criteria in the integrated framework, it will increase the awareness and confidence about the security software and Common Criteria certification among users.

**ACKNOWLEDGMENT**

Utmost gratitude and appreciation to those who have been directly or indirectly involved in the preparation of this project report. In particular, I would like to extend special thanks to my supervisor, Dr. Mohd Naz'ri Mahrin

**REFERENCES**

[1] Adams B, Schutter KD and Zaidman A (2009). The Journal of Systems and Software Using aspect orientation in legacy environments for reverse engineering using dynamic analysis. *The Journal of Systems & Software*. 82(4):668-684.

[2] Alonso F, Juristo N and Mat JL (1996). Software Engineering and Knowledge Towards a Common Life Cycle. *Structure* 1212:65-79.

[3] Antonellis P, Antoniou D and Kanellopoulos Y (2009). Code4Thought Project: Employing the ISO/IEC-9126 Standard for Software Engineering-Product Quality Assessment. *2009 13th European Conference on Software Maintenance and Reengineering*. 297-300.

[4] Aoyama M (1996). Beyond software factories: concurrent-development process and an evolution of software process technology in Japan. *Information and Software Technology*. 38(3):133-143.

[5] Ardi S and Shahmehri N (2009). Introducing Vulnerability Awareness to Common Criteria's Security Targets. *2009 Fourth International Conference on Software Engineering Advances*.419-424.

Fig.

[6] Begel A and Nagappan N (2007). Usage and Perceptions of Agile Software Development in an Industrial Context: An Exploratory Study.

[7] First International Symposium on Empirical Software Engineering and Measurement (ESEM 2007). 255-264.

[8] Bimler D, Kirkland J and Pichler S (2004). Escher in color space: individual-differences multidimensional scaling of color dissimilarities collected with a gestalt formation task. *Behavior research methods, instruments, & computers : a journal of the Psychonomic Society, Inc.*36(1):69-76.

[9] Bond GW, Cheung E and Goguen HH (2005). Experience with Component-Based Development of a Telecommunication Service. *Development*. 298-305.

[10] Briand LC, El Emam K, Surmann D, Wiczorek I and Maxwell KD (1999). An assessment and comparison of common software cost estimation modeling techniques. *Proceedings of the 21st international conference on Software engineering - ICSE '99*. :313-322.

[11] Bru F-xavier, Frappin G, Legrand L and Merrer E. (2001) Building an Observatory of Course-of-Action in Software Engineering : Towards a Link between ISO / IEC Software Engineering Standards and a Reflective Practice. :185-200.

[12] Cheikhi L, Abran A, Suryn W, Superieure EDT and Ouest N-dame (2006). Harmonization of usability measurements in software engineering standards. *Cycle*. 3246-3251.

[13] De Win B, Scandariato R, Buyens K, Grégoire J and Joosen W (2009). On the secure software development process: CLASP, SDL and Touchpoints compared. *Information and Software Technology*. 1(7):1152-1171.

[14] Dong J, Wang J, Sun D and Lu H (2008). The Research of Software Product Line Engineering Process and Its Integrated Development Environment Model.

[15] Drehmer D (2001). A note on the evolution of software engineering practices. *Journal of Systems and Software*. 57(1):1-7

[16] Dyba T and Dingsoyr T (2008). Empirical studies of agile software development: A systematic review. *Information and Software Technology*. 50(9-10):833-859.

[17] Dyba T, Kampenes V and Sjøberg D (2006). A systematic review of statistical power in software engineering experiments. *Information and Software Technology*.48(8):745-755.

[18] Eia (1998) I. IEEE / EIA Standard International Standard Technology — Software life cycle processes. *Electronics*.

[19] Eia (1995) I. IEEE / EIA Guide Industry Implementation of International Standard ISO / IEC 12207 : 1995 ( ISO / IEC 12207 ) Standard for Information Technology — Software life cycle processes — Life cycle data. *Source*.

[20] Emmerich EW. Chapter 2 Software Process Improvement Standards , Assessments and. *Engineering*.

[21] Engberts A, Ning JQ and Drive SW (1994). Transferring re-engineering technology to a software development and maintenance organization: an experience report. *Proceedings International Conference on Software Maintenance ICSM-94*. 100-108.

[22] Engelsma ES (2000) Incremental Systems Integration within Multidisciplinary Product Line Engineering Using Configuration Item Evolution Diagrams. *Mechatronics*.

[23] Finnie GR and Wittig GE (1997), Desharnais J-m. Estimating Software Development Effort with Case-Based Reasoning. *Analysis*.

[24] French VA and To J (1995) , eFacv Defence Svstem M-ce : d Pmcess Imgmvmea -rice RePr [ Ilf. *Sciences-New York*.

[25] Goulao M and Brito e Abreu F (2007). Modeling the Experimental Software Engineering Process. *6th International Conference on the Quality of Information and Communications Technology (QUATIC 2007)*. 77-90.

[26] Habra N, Alexandre S, Desharnais J, Laporte C and Renault a (2008). Initiating software process improvement in very small enterprisesExperience with a light assessment tool. *Information and Software Technology*. 50(7-8):763-771.

[27] Han J and Zheng Y (2000). Security characterisation and integrity assurance for component-based software. *Proceedings International Conference on Software Methods and Tools*. SMT.61-66.

[28] Hwang S-myung (2004). A Design of Configuration Management Practices and CMPET in Common Criteria Based on Software Process. *Interface*.15504:481-490.

[29] Irish A, Vses S, Study C, Basri SB and Connor RVO (2010). Organizational Commitment Towards Software Process Improvement. *Assessment*.1456-1461.

- [30] Keglawi F and Sullivan D. Applying the common criteria in systems engineering. *IEEE Security & Privacy Magazine*. 4(2):50-55
- [31] Lee J, Lee J, Lee S and Choi B (2003). A CC-based security engineering process evaluation model. *Proceedings 27th Annual International Computer Software and Applications Conference. COMPAC 2003*. 130-135.
- [32] Maletic JI, Howald a and Marcus a (2003). Incorporating PSP into a traditional software engineering course: an experience report. *Proceedings 14th Conference on Software Engineering Education and Training*. "In search of a software engineering profession" (Cat. No.PR01059). 89-97.
- [33] Management WS, Site W and Cycle L (2006). INTERNATIONAL STANDARD ISO / IEC Software Engineering — Recommended Engineering , Web Site Management , and. *Electronics*.
- [34] Mark L and Rombach HD (1989). Generating customized software engineering information bases from software process and product specifications. [1989] *Proceedings of the Twenty-Second Annual Hawaii International Conference on System Sciences. Volume II: Software Track*. 587-595.
- [35] Mason J (2003). Aligning workforce development & software process improvement strategy for accelerated adoption of software engineering capability. *Proceedings 16th Conference on Software Engineering Education and Training, 2003. (CSEE&T 2003)*.70-77.
- [36] Mellado D, Fernandezmedina E and Piattini M (2008). Towards security requirements management for software product lines: A security domain requirements engineering process. *Computer Standards & Interfaces*. 30(6):361-371.
- [37] Mellado D, Fernandezmedina E and Piattini M (2007). A common criteria based security requirements engineering process for the development of secure information systems. *Computer Standards & Interfaces*. 29(2):244-253.
- [38] Mellado D, Blanco C, Sánchez LE and Fernández-Medina E (2010). A systematic review of security requirements engineering. *Computer Standards & Interfaces*. 32(4):153-165.
- [39] Mellado D, Fernandez-Medina E and Piattini M (2008). Security Requirements Variability for Software Product Lines. *2008 Third International Conference on Availability, Reliability and Security*. 1413-1420.
- [40] Mellado D, Fernández-Medina E and Piattini M (2010). Security requirements engineering framework for software product lines. *Information and Software Technology*. 52(10):1094-1117.
- [41] Montoni M, Santos G and Rocha AR (2006). Taba Workstation : Supporting Software Process Deployment Based on CMMI and MR-MPS . BR. 2006:249 - 262.
- [42] Morali A, Zambon E, Houmb SH, Sallhammar K and Etalle S (2009). Extended eTVRA vs. security checklist: Experiences in a value-web. *2009 31st International Conference on Software Engineering - Companion Volume*. 130-140.
- [43] Rabbi F, Wang H and MacCaul W (2010). YAWL2DVE: An Automated Translator for Workflow Verification. *2010 Fourth International Conference on Secure Software Integration and Reliability Improvement*. 53-59.
- [44] Rao AS (2000). TECHNOLOGY ACCEPTANCE MODEL FOR COMPLEX TECHNOLOGIES IN A PERIOD OF RAPID CATCHING-UP.
- [45] Scacchi W (1999). Experience with software process simulation and modeling. *Journal of Systems and Software*. 46(2-3):183-192.
- [46] Seffah A, Donyaee M, Kline RB and Padma HK (2006). Usability measurement and metrics: A consolidated model. *Software Quality Journal*.14(2):159-178.
- [47] Sherif K (2002). Domain engineering for developing software repositories: a case study. *Decision Support Systems*. 33(1):55-69.
- [48] Siau K and Tan X (2005). Improving the quality of conceptual modeling using cognitive mapping techniques. *Data & Knowledge Engineering*. 55(3):343-365.
- [49] Skandan S and Sidhardhan M (2009). Software Engineering Education at TCS Induction - An Experience Report. *2009 22nd Conference on Software Engineering Education and Training*. 16-19.
- [50] Stallinger F, Dorling A, Henderson-sellers B, Box PO and Lefever B (2002). Software Process Improvement for Component-Based Software Engineering. *An Introduction to the OOSPICE Project The Practice of Component-based Software Engineering*. (Section 2).
- [51] Standard I (2007). INTERNATIONAL STANDARD ISO / IEC. *October*. 2007;2007.
- [52] Standard I (2008). INTERNATIONAL STANDARD ISO / IEC. 2008;2008.
- [53] Tompkins F (1986). Integrating security activities into the software development life cycle and the software quality assurance process. *Computers & Security*.5(3):218-242.
- [54] Trammell C (1996). The incremental development process in Cleanroom software engineering. *Decision Support Systems*. 17(1):55-71.
- [55] Tripp L (1987). Taxonomy of software engineering standards: A development history. *Computer Standards & Interfaces*. 6(2):195-205.
- [56] Tsai JJP and Liu A (2009). The Journal of Systems and Software Experience on knowledge-based software engineering : A logic-based requirements language and its industrial applications. *The Journal of Systems & Software*. 82(10):1578-1587.
- [57] Upchurch RL and Sims-knight JE (2002). Session S2G PORTFOLIO USE IN SOFTWARE ENGINEERING EDUCATION : AN EXPERIENCE REPORT Session S2G. *Education*. 2-6.
- [58] Ushakov I, Werling R and Nagano H (1993). Graphical representation of information in software engineering: ISO-IEC JTC1/SC7 framework, concepts and standards activity. *Proceedings 1993 Software Engineering Standards Symposium*. 209-218.
- [59] Valett JD and MCGarry FE. A S (1989). Software Measurement Experiences in the Software Engineering Laboratory. *Personnel*.148:137-148.
- [60] Verner JM, Sampson J, Tosic V, Bakar N a A and Kitchenham B a (2009). Guidelines for industrially-based multiple case studies in software engineering. *2009 Third International Conference on Research Challenges in Information Science*. 313-324.
- [61] Walker A (1998). Improving the quality of ISO 9001 audits in the field of software. *Information and Software Technology*. 40(14):865-869.
- [62] Wang Y, King IG and Dorling A (1998). A Worldwide Survey of Base Process Activities Towards Software Engineering Process Excellence. *Society*. 1-4.
- [63] Wangenheim C, Weber S, Hauck J and Trentin G (2006). Experiences on establishing software processes in small companies. *Information and Software Technology*. 48(9):890-900.
- [64] Ware MS, Bowles JB and Eastman CM (2006). Using the Common Criteria to Elicit Security Requirements with Use Cases. *Proceedings of the IEEE SoutheastCon 273-278*.
- [65] Zainol A and Mansoor S (2008). Investigation into Requirements Management Practices in the Malaysian Software Industry. *2008 International Conference on Computer Science and Software Engineering*. 292-295.