

# Research Whitepaper Outline: Software Models, Architecture and their Maintainance attributes

Gautam Kumar  
Instructor: Prof. Mark Kochanski

January 24, 2016

## 1 Introduction

Software Maintainability as a quality attribute can be simply defined as “The easiness of maintaining a software system.”[5]. The actual process of maintaining software systems is affected by a lot of related factors such as Analysability, Changeability, Testability and Adaptability as described by the ISO/IEC 25010:2011 standard. The goal of the proposed white paper is to analyse existing literature and propose metrics for evaluating software designs and architectures.

## 2 Whitepaper Outline

1. Introduction
  - (a) Problem Statement
  - (b) Relevance and reasoning
  - (c) Definition of uncommon / ambiguous terms
2. Software Maintainance
  - (a) Describe Software maintainance and maintainability.
  - (b) Define attributes which affect software maintainance.
  - (c) Define metrics to quantify maintainability and its related attributes
3. Analysis of Software Models
  - (a) Describe a few common Software Models

- (b) Apply metrics calculations on Software Models
  - (c) Analyse and interpret results.
- 4. Analysis of Software Architectures
  - (a) Describe a few common Software Architectures
  - (b) Compute maintainability metrics of common software architectures
  - (c) Analyse and interpret results.
- 5. Conclusion
  - (a) Conclusions
  - (b) Recommendations if any

### 3 General Direction

The general focus of the proposed white paper would be to analyse and evaluate existing Software Models and architectures using maintainability metrics to discover patterns which can be used to reliably predict the maintainability of a proposed software model or architecture.

Some research papers of interest are

- “Distributed Scrum” [4]  
This paper talks about how the Agile model helped a globally distributed team build a large software project without compromising on important quality attributes such as Maintainability, Testability while still maintaining sufficient productivity.
- “A Systematic Review of Software Maintainability Prediction and Metrics” [3]  
This paper evaluates various methods of predicting software maintainability for their effectiveness.
- “Maintainability prediction: a regression analysis of measures of evolving systems” [2]  
This paper proposes a method to predict the maintainability of various software systems.
- “Building UML class diagram maintainability prediction models based on early metrics” [1]  
UML class diagrams can be a useful way to predict the maintainability of a early software model / architecture by analysing the complexity and modifyability. This paper offers us a method to predict maintainability of such UML class diagrams.

## Bibliography

- [1] GENERO, M., MANSO, E., VISAGGIO, A., CANFORA, G., AND PIATTINI, M. Building measure-based prediction models for UML class diagram maintainability. 517–549.
- [2] HAYES, J., AND ZHAO, L. Maintainability prediction: a regression analysis of measures of evolving systems. In *Proceedings of the 21st IEEE International Conference on Software Maintenance, 2005. ICSM'05*, pp. 601–604.
- [3] RIAZ, M., MENDES, E., AND TEMPERO, E. A systematic review of software maintainability prediction and metrics. In *Proceedings of the 2009 3rd International Symposium on Empirical Software Engineering and Measurement, ESEM '09*, IEEE Computer Society, pp. 367–377.
- [4] SUTHERLAND, J., VIKTOROV, A., BLOUNT, J., AND PUNTIKOV, N. Distributed scrum: Agile project management with outsourced development teams. In *40th Annual Hawaii International Conference on System Sciences, 2007. HICSS 2007*, pp. 274a–274a.
- [5] ZHU, H. *Software Design Methodology : From Principles to Architectural Styles*. Butterworth-Heinemann.