Conclusion
9.0 Conclusion and Future Work

This thesis has explored a wide range of highly related issues with respect to the aging of application software and software in general. With relation to the capacity and stability of available concepts and principles, reviewed literature has thus shown that Application Software Reengineering is a very narrowly researched area regardless of its indispensability in the present day business; corporate and scientific domains, were a products' time to market is inherently important inasmuch as its functionality.

Forward and Reverse engineering techniques are always mentioned whenever application software reengineering is mentioned. However, we have also sought to address this anomaly by highlighting that its more important to have standard process models and or frameworks governing the discipline as opposed to merely mentioning the tools.

To this we now present an overall conclusion and summarise our contributions. Further, we point out the limitations and identify specific directions for future work.

9.1 Summary of our Contributions

Throughout this thesis, we have explored and closely followed the main reason why and how application software ages overtime. We have taken a gently skeptical approach, accepting and building on existing concepts and principles we have built a new more robust framework for extracting variable names from source code. The following contributions and conclusions may therefore be put forward:
• We have developed a complete framework to explore source code files for variable names and their attributes;
• We have proposed several steps to be followed from extraction, cleaning, matching and storing the variable name;
• As part of our Framework to explore variable names from source code files, we have also gone through several Software engineering principles inasmuch as Software Reengineering Concepts and methodologies.

9.2 Limitations

In our approach, we clearly ignore the effect of Grammar and Architecture of the System as an Enterprise Solution. We have assumed that grammar and Architecture are not altered; only additional classes and variable names were added, subtracted or updated due to new business rules or change thereof. But no major rework has been conducted on the Grammar inasmuch as Architecture.

However, where the grammar and the Architecture have been altered then a framework for understanding the Grammar and Architecture and a comprehensive follow-up of the same over a period of time has to be taken into consideration and mitigate all of them to come up with a comprehensive and wholesome Framework.