**3. Research Methodology**

This chapter discusses the systematic approach to be performed in order to accomplish the objectives of this research.

**3.1 Software Development Process**

A variant of the Waterfall model, the Modified Waterfall model, seen in Figure 3.1, will be followed. The Modified Waterfall model includes a feedback mechanism for evaluating and validating the output of each phase in the software development lifecycle. Its overlapping stages facilitate the modification and improvement of any aspect of the system immediately, from requirements specification, architectural design and detailed design to implementation and testing. It ensures that the milestones have been accomplished, and it provides flexibility in updating the output of previous stages when necessary.

**3.2 Software Concept**

During this stage, the research topic is defined and conceptualized. Related systems and papers on related concepts will also be gathered and reviewed. Consultations with the thesis adviser will be conducted regularly to monitor the progress of the project, to evaluate the current state of the technology in the country, and to decide on project objectives and scope.

During consultations, comments and suggestions will be considered. The omissions indicated on submitted documents will be immediately reassessed and changed, while the suggested readings will be read. Also, similar systems will be reviewed to fully understand and familiarize the depth of the research topic. Further research on conceptual relations and the different types will be done.

**3.3 Requirements Analysis**

During this stage, data gathering and research will be performed to identify the following: types of conceptual relations, architecture of relation extraction systems, algorithms for extracting conceptual relations. Additionally, the input corpus consisting of at least 30 children’s stories will be gathered. The suitable programming language for the project will also be identified. Furthermore, interviews with Filipino language professors and linguists will be conducted.

After gathering the data, the requirements will be defined and analyzed to determine the objectives and scope of the system. The resulting requirements specification will be validated to ensure completeness of the software. Moreover, Use Case diagrams will be developed to help guide in creating the software by determining the appropriateness of the different functions in the system.

**3.4 Architectural Design**

In the Architectural Design stage, the overall function and subsystems will be identified. Also, relationships, dependencies, and interactions among the subsystems will be determined. The resulting architectural design will be reviewed to help understand the flow and design of the system more.

**3.5 Detailed Design**

At this stage, the classes and functions will be defined. The data structures and libraries to be used in the system will also be identified. The extraction templates for the different conceptual relations will be done as well. Furthermore, algorithms for extracting conceptual relations will finally be formulated.

**3.6 Coding and Development**

The actual implementation of the software design will be done in the fifth stage, Coding and Development. Debugging and unit testing will be done regularly to ensure the efficiency and correctness of the software.

**3.7 Testing**

Testing will be done to ensure the quality and efficiency of the software. Unit testing for each subsystem will be performed. After doing so, integration testing will be performed to verify that each subsystem receives the correct input from the previous subsystem and generates the appropriate result for use by subsequent subsystems.

Test cases will be employed to check that all subsystems interact correctly. System and functional testing will also be performed to check the functionality and performance of system functions. Lastly, the outputs of the system will mainly be evaluated through the use of PictureBooks. The generated story of PictureBooks after using the output semantic network will be evaluated. The output semantic network may also be evaluated by Filipino linguists to ensure their validity.

**3.8 Documentation**

Throughout the entire process of developing the software, documentation will be done to track the progress of the software. This is also to ensure that any changes and implementations in the requirements of the software will be reflected in the documents.