Chapter 3: Methodology

3.1 Methodology

Methodology covers the methods used to discover past, existing and forthcoming new technologies in research and developing a new project. It is a collection of tools and techniques used to aid system developers in the development of a new system, including the proper procedures used to study existing systems with the aim of eliciting improvements to current technology.

These tools and technology differ from discipline to discipline. There are qualitative and quantitative approaches to arriving at a solution. Researchers have their own biases, depending on their training and preferences. A quantitative approach provides a rich resource for interpretation. Some prefer qualitative over quantitative approaches. An integrated approach is the best option. This project tries to mix the best of both methods.

Research methodology comes loosely under the following approaches:

- **Research Ethics**
  It is not ethical for research to copy wholesale from existing research findings, although it is proper to base the research upon various research papers. However, the research should always include new findings discovered by the researcher.

- **Secondary Data Sources**
  Conducting Interviews and surveys are time-consuming. Often it is impossible to conduct such research in a local area. This is where the experience of other researches can be drawn upon. The results of multiple surveys done by previous researchers can be utilized effectively.
• **Qualitative methods**

These methods include Action Research, Content Analysis, Ethnography, Narrative Inquiry and many others. These are generally arts-based methods that require descriptive explanations of various happenings and events. However, it could include scientific qualitative explanations of phenomena or concepts in technology.

• **Quantitative Methods**

These methods refer to the use of statistical analysis of the results of surveys. The idea is to draw conclusions or inferences about a certain pattern of behavior. Such methods are useful for testing the functionality and performance of projects.

• **Textbook Research**

Much information can be gathered from textbooks which cover important concepts related to an area of research. This provides a valuable source of information that is reliable and dependable.

• **Internet research resources**

This is a vast resource, and in recent years, the amount of information that can be gathered has increased exponentially. A huge quantity of information can be obtained from various search engines, chief among recent ones are Google and Yahoo.

This project uses methodology that involves building of a software product. Building computer software is an iterative learning process, and the outcome is an embodiment of knowledge collected, distilled, and organized as the process is conducted (Pressman, 2001). There is much to be learned, and the sum total of the knowledge aids in the development process. This involves the use of a software model to aid in the design and development of the whole software product.
3.1.1 Software Process Models

There are various software process models that serve as an abstract representation of the software process. These include:

- **The Waterfall Model**
  This involves projects that have well-defined phases of specification, software design, development, implementation, validation and evolution. Each of these phases follows one after another.

- **Evolutionary development**
  This approach is based on the idea of developing an initial implementation, exposing a prototype to the user, obtain feedback, and revising the system until it meets the requirements of the end-user. This is useful for projects where the end-user requirements are not clear at the beginning of the software process.

- **Formal systems development**
  A formal mathematical system specification is first produced, and then transformed using mathematical methods. Verification of the various components involves mathematical techniques.

- **Reuse-based development**
  The trend in recent software development is component technology, which fosters the re-use of existing components to build newer and larger components.

3.1.2 Choice of Reuse-based development

By looking at the above software process models, this project has taken a Reuse-based development, coupled with some evolutionary steps in the final stages of development. The Reuse-based approach is especially useful when developing Web Services, as the very existence of Web Services involves the concept of programmable software components that can be re-used by other programs developed by third-party software programmers.
This software model involves the following phases of development as shown in Figure 3.1.

![Diagram showing the phases of software development]

Figure 3.1 Reuse-based Software Development Process

Much of the software is based upon components that have been designed during previous software development projects. The components developed using earlier projects were not true re-usable components in the sense that they use component technology such as COM and DCOM. The earlier components were nothing more than software modules and functions built using the C and C++ programming language. However, this project will select the C# language, which uses syntax which is similar or even identical to C/C++ function modules. These can be re-used in modules developed using the C# programming language, since there are many similar code modules written in the C language that can be re-used in the C# language. This re-use capability is one of the strong reasons for the choice of the C# language. The six phases in the reuse-based
software development process are described in detail below.

3.1.2.1 PHASE 1 - Requirements specification

The first phase of this development involves the Requirements specification, which is the usual starting phase of all the software process models. We need to determine what is required to be developed at the beginning phase of the project. Referring to the users of the system sets out the system’s services, constraints and goals. These requirements are described and defined in detail, serving as the system specification.

The full details of the requirements specification are described in Chapter 4. These involve the functional requirements, the non-functional requirements and the hardware requirements. The methodology used in gathering these requirements involves several channels of requirements discovery. These sets of requirements are based upon extensive use and development of examinations systems over the past decade. Previous development was conducted in an ad hoc manner, where requirements were met as the need arose. Some of the requirements were gathered through user feedback and user requests conducted in an informal manner. Other requirements were the result of past constraints on the database system capabilities. There are requirements that resulted in evolving management goals and expectations, some from the school management, while others are requirements imposed by the Education Department of the Ministry of Education. This project collects all these requirements together in a more formal manner, in effect reusing these past requirements, and adding new requirements as the system evolved.

The main goal of these requirements is to develop a system that enables the teachers, together with assistance from their class monitors, to enter the school’s internal examinations results into a system with a user interface that is easy to use. The system should be able to make full use of the capability of the school Intranet System. The results should be processed using the processing power of computers, and the reports should give a good idea of the performance and trend of the schools examination results.
3.1.2.2 PHASE 2 - Component analysis

The next stage involves identifying and searching for those components that have already been developed. These components have been developed earlier using the C/C++ programming language. Some modifications are necessary because of some differences in which the C# language is used. Much of this part of the project is dependent upon the knowledge of examination requirements of the public as well as internal examinations. There are functions that involve the method of calculating averages, which depend on certain criteria that must be met; it is not a simple mathematical average. This pool of knowledge has been acquired over many years of development leading to the current use of Web Services. This is where re-use of these components are vital for a speedier development of this project. In a world where software has matured, users expect a certain degree of completeness and quality that they have been accustomed to when using earlier software. Re-use of these matured components are vital to the success of modern software.

The main components that can be reused can be classified as follows:

- The methods for calculating averages.
- The combination of subjects that form the average figure. These can vary from class to class and the earlier components can identify these combinations. The combination of subjects has been fluctuating through the years, as requirements for subject packages offered to the students change as the requirements of the Education Department changed. Over the many years, several methods of calculating averages have been used, discarded, and reused.
- The methods for calculating the position in class. There were methods that involve use of a combination of average marks, aggregate marks, level achievement. These methods have changed several times over the years, and knowledge of these methods is reused in designing the various functions for calculating position in class, and position in form.
The overall process flow in collecting the marks and the methods to process these marks.

3.1.2.3 PHASE 3 - Requirements Modification

In the Requirements Modification phase, it is necessary to analyze and identify those components that need to be modified. In EWSS, the components that need to be modified include the following areas:

- Shift from C-language to C#-language
- Shift from DOS-based function calls use .NET framework classes
- Shift from single-user to client-server technology
- Shift from stand-alone application to use Web Services
- Shift from file-based database technology to relational database management system

In this case some of the functions need to be modified to conform to the specifications of the new language, which is C#, even though many of these differences are minor. Shifting from C-language to C#-language involves using of classes compared to just plain function-calls. Apart from organizing the structure of the program into classes, many of the internal functions were a simple re-use of the C function structures that were used in the earlier systems. However, many these functions needed minor modification.

There is a change in the concept of the operating system from DOS-based programming to Windows-based programming, especially in the .NET Framework mindset. Much of the modification phase takes into consideration of the use of components, in fact re-use of Windows and .NET Framework components. The user interfaces are now forms-based and Web-based as compared with text-based interfaces.

The older system uses a single-user concept. The client-server concept must consider factors that involve several users accessing the same database at the same time, where much thought has to be put into the re-design of some of the components,
especially the database components.

It is also necessary to modify these components with the view of re-use as it fits the model of Web Services. There is a new paradigm in the use of Web Services that were not thought of when designing the earlier components. Instead of embedding functions within a single executable file, the functions have been modified to act as Web Service calls that operate within the framework of the Web Server. The Web Server is now the provider of these services which the client calls as a Web Service. We are re-using some of the tried and tested functions. The difference in many cases lies in the locality of the function.

Also, we have changed the whole database access technology, including the actual database itself. The earlier components have used the dBase file-based approach to database access. The current project now uses SQL Server, and ADO.NET database access methods. Many of the earlier components must be modified to conform to the new database access methods. Especially pertinent to this discussion is the shift from record-based, and recordset-based database access to the use of Datasets. So is the use of SQL statements compared to direct manipulation of database fields and records? However, the concepts regarding which database fields to access are dependent on previous knowledge of these requirements.

3.1.2.4 PHASE 4 – System Design with reuse

The next phase leads to System design with reuse. This involves a combination of components that can be re-used, components that need to be modified, new components that need to be developed, and discarding some old components that are too laborious to modify. The new system involves the use of newer technologies which are often much easier to program, eliminating elaborate work-around for some old routines. These are combined together into a new framework, which in many ways resemble the old framework.

In Phase 2, we identified the components that can be re-used. In Phase 3, we
identified the components that need to be modified. Here in Phase 4, we need to identify the new components that need to be developed, and then combine them all into one single, integrated framework.

The new components that need to be developed are identified as follows:

- Design and implementation of the new database management system
- Setting up a Web Server
- Design of the Web Service architecture
- Design of the user client interfaces
- Design of new methods of processing the examination results data

A new database needs to be designed and implemented. This is designed as a Microsoft SQL 2000 Server database, named EWSS1DB. The details are covered in Chapter 5.3.

The Web Service architecture involves the various services such as Web Services, Database Services, and Clients Services. These are covered in detail in Chapter 5.1.

The client interfaces involves the data-entry module, the data-processing module, and the data-analysis module. These are covered in Chapter 5.4.

One of the new methods that need to be developed is the calculation of the grade-point average, called the "Gred Purata Mata Pelajaran", or GPMP in short. This will be debated through later chapters of this thesis.

3.1.2.5 PHASE 5 - Development and Integration

The development and integration phase involves the actual developing of those components that need to be developed, modifying components that need to be modified, and integrating with existing components. This is one of the most critical phases as the failure to develop new components will result in a collapse of the whole project. New training skills are required, which have to be learned or developed.
While Phase 4 identifies the components that need to be developed, Phase 5 develops and implements all the design requirements identified in the earlier phases. This is where the actual development begins, and the new components developed. This is the where much of the programming in the new language, the C#-language, begins.

The development involves the implementation of the EWSS Web Services Architecture as described in detail in Chapter 6. This phase involves the installation of the .NET Framework, the SQL Server, and the programming environment, which is the Microsoft Visual Studio.NET.

In order to implement this phase, new skills have to be developed. Specifically, this means learning the .NET Framework, the C# language and how it implements SQL functionality.

3.1.2.6 PHASE 6 - System validation

The final stage in this process model is common to all the different process models. The product needed to be tested for conformance with the system requirements as well as assessed as to its suitability for the task of processing examination results, that is, the overall goal of the project. This is the validation stage.

At this phase, user involvement is vital to the design, redesign and validation of the user interfaces. Much input has been gathered from the users as to the design of the various data entry screens, processing screens and analysis screens. User input was vital in determining the type of reports that were generated. In this case, it involves the feedback from the users concerning the printing of mark sheets and analysis reports. Again, in this case, there was much reuse of the previous reports in terms of format and the information that these reports can provide.

There was much debate concerning some new methods of calculating the position in class. This debate will be covered in detail in Chapter 7. The debate arises from the validity of the Grade Point Average method for determining the position in class. Such
validation procedures are necessary for ensuring the integrity of the methods used in the system.

3.1.3 The Process Iteration

There are two approaches to process iteration, where, during the evolution of system requirements, parts of the process need to be repeated.

The first approach is incremental development, where software specification, design and implementation are grouped and divided into smaller parts as a series of increments, each developed in turn.

The second approach is spiral development, where the development spirals outwards from where it began, and finally reaching the final stage.

This project takes on the re-use software process model with incremental development process. In this approach, it is necessary to have an overview of the goal of the project to be kept in mind all the time. It is then necessary to identify those parts and components that need to be studied or developed early in the development. In this EWSS project, it was necessary to develop and test those areas that are crucial to the final goal, that is, to be able to process results, print mark sheets, and produce intelligible analysis reports. These are grouped into two broad categories:

- Technology functionality increments
- Component functionality increments

Technology functionality involves literature review, choosing the right technology, and then testing the suitability of the technology chosen. These technologies include Web Services, the SQL Server capabilities, and the ADO.NET database access methods.

It was necessary to know if Web Services serve the goal of processing of examination results. Then it was necessary to know if Web Services will function, given
the equipment that is available. The computers available in the computer room consist of a mix of older and newer hardware, together with various versions of the Windows operating system.

If Web Services functions, the next increment will be to develop components for database access using the chosen access method, that is, ADO.NET, to access the data in a database created in SQL Server 2000.

Only when it was found that these increments would function that the next series of increments were designed. This next series is the component functionality, which forms the bulk of the development increments. This involves incremental development of the various components within the EWSS architecture, as described in detail in Chapters 5 and 6.

Hence the incremental development phases are shown in Figure 3.2 and Figure 3.3.

![Diagram showing the two stages in overall functionality increments.](image-url)

*Figure 3.2 Two stages in overall functionality increments*
3.2 Techniques Used to Define Requirements

During the methodology stage, various techniques are used to define the requirements of the system. These techniques helped in formulating the various requirements of the system, as detailed in the next chapter. These techniques include the following:

- Analysis of existing systems
• Feedback from users of existing systems
• Review of new technologies
• Library Research
• Purchase of new books on relevant topics
• Internet Browsing

3.2.1 Analysis of existing systems

It is vital for the success of a new project to learn from the strengths and weaknesses of existing systems. Studying the existing systems in fact resulted in the definition of the bulk of the requirements of the system. From examining the code of the existing system, we can trace how all the functions are designed. By using and testing these systems, it was possible to determine what the requirements that must be included in the system. Through this method, many of the bugs in the system were discovered, and from code analysis, the source of these bugs was determined. However, this project did not attempt to rectify those bugs and weaknesses, as it was determined that the new system would replace the old system.

3.2.2 Feedback from users of existing systems

From the comments received from users of the system, it was possible to determine the performance and pitfalls the examination processing system. This is important because through studying feedback from many types of users, whether people who are skillful in the use of computers, or whether people who are novices in computers, it was necessary to modify requirements to meet the usage pattern of a wide variety of people. It was also possible to build a 'wish-list' of requirements based upon feedback from these users. These would form the user-centered list of requirements.

The demography of the users consists of the following groups of users based upon their job functions:

• School Administrator (Principal, Senior Assistants)
• Department Head
- Examinations Administrator
- Clerical Staff
- Subject Teacher
- Form Teacher
- Student

Each of the above categories of users views the system in different ways. The school administrators are primarily concerned with the final mark sheets and analysis reports. The department heads view the overall analysis related only to their respective departments. The examinations administrator is concerned with generating the analysis reports and the mark sheets. The clerical staff views only the mark sheets for preparation of forecast results. The subject teachers are interested only in the performance of their own subjects. The form teachers look at the mark sheets of their own class. The students play a vital role in assisting the form teachers for data entry. Some form teachers are not computer literate enough to handle data entry by themselves.

There are various sources of user feedback, which can form the basis of various system requirements. These include verbal requirements requests, where teachers and administrators convey the types of reports they need, or would like to include into the system. Minutes of staff meetings that record needs of examination requirements provide some insight into some of the user requirements. Circulars received from the Education Department contain certain changes to the methods of reporting the school results have created the need for changes to the system to meet new requirements. The most recent of these requirements involve the calculation of the Grade Point Average for each subject and for the school overall results. One-to-one meetings with school administrators often bring out many requirements that help the administrators view the overall trend of the examination results.

Questionnaires are generally difficult to administer in a school environment. Teachers generally are too busy with their teaching schedules to conscientious write out their needs on paper. It is generally easier to have informal interviews with them to
extract their views on the system.

The role of questionnaires lies in getting their feedback on the user interface and the implementation of the system. This was done with a preliminary questionnaire that allows freely structured comments. This forms the basis of the final questionnaire that is appended in Appendix A.

The methodology of the analysis of the questionnaire consists of the following steps:

- Prepare the questionnaire and print (100 copies)
- Explain the questionnaire to the users
- Issue the questionnaire to the users
- Collect the responses from the users
- Tally the results in a tally form designed and programmed in C# in the .NET Framework¹
- Import the output into Microsoft Excel
- Insert charts to visualize the data
- Summarize and analyze the results

The analysis of the questionnaire on the feedback from users is shown in Chapter 8, while the questionnaires and detailed charts are appended in Appendix A.

3.2.3 Review of new technologies

By reviewing and understanding the new features and capabilities of new technology, it was found that there are requirements that could be included which otherwise would have been deemed impossible. This is most often done by reading computer books, journals, magazines and articles. Sources can be obtained from the Internet and book shops.

¹This is a small program designed to facilitate tallying of results. It generates a text file that can be imported into Microsoft Excel.
3.2.4 Library Research

The Library at the University of Malaya stocks many books of inestimable value to the understanding of some of the technology used to develop this project. These books could either be browsed or borrowed, resulting in a change in some of the requirements of the system.

However, books in libraries on new technologies are often outdated, as libraries do not generally purchase new books as fast as they are published.

3.2.5 Purchase of new books on relevant topics

The Library does not stock many important titles that prove vital to kick-start this project. These books had been purchased. As with all computer texts, many of these books become outdated very quickly. Often a book would be bought, and it was found, through the Internet, that a new edition had been published. This happened especially to the books on the latest technologies, in particular, the .NET Framework. So it was necessary to check the latest edition of a particular title from the Internet, before embarking on the purchase of a new book.

Some bookshops such as MPH have reading corners to browse through their collection of books. This is the best source of the latest available books on new technologies. Some books are too expensive to be purchased. With so many new titles being published, it is not within budget to buy all available books in the market.

3.2.6 Internet Browsing

This is a must, as much information in the form of reviews of Technology can be found in the web sites of the vendors of the software or books. There are many tutorials, and trial kits that are available for download from the Internet. White papers from various vendors are valuable information that often can be downloaded for free. Just plain Internet browsing and use of search engines such as from Yahoo and Google have proven

---

1 The reference section lists some of these books.
to result in vital information which could not be found in books. Through the search engines, it was possible to find methodology for research on topics useful for the development of this project.

It is also possible to purchase the latest publications on various topics of research through the Internet. Amazon.com and InformIT.com are two good sources of books. So are the web sites of various publications such as McGraw Hill and Wrox.com.

3.2.7 Subscription to e-magazines

There are many electronic magazines on various topics such as Search Database.com and Search Web Services.com that deliver valuable information on a regular basis, up to on a daily basis. These electronic magazines have links to the valuable Web Sites containing tutorials, tips which have greatly assisted in determining technology features which helped in determining requirements and the development process itself.

There are even some vendors which send their hardcopy magazines without charge upon a free registration. These include IBM DeveloperWorks, Oracle and DB2 Magazine. One such source was a free copy on an issue covering Web Services, which we received, giving useful information for literature review and the modification of some of the system requirements (developerWorks Journal, Nov 2002). This is referred to in the Reference Section of this dissertation. However, some of these magazines are available only to residents of the United States of America.

3.2.8 e-Books, that is, electronic books.

Many publishers have made some of their books available as e-books that could be downloaded from the Internet. Some publishers such as Wrox Press even provide free membership to some of their resource materials on a restricted basis. Many have made some free chapter downloads, some of which were found to be extremely useful in the

---

3 They send emails regularly on topics which are very useful for developing Web Services. One can sign up for free membership to receive these emails.
development of this project. They even have paid membership that can access a larger variety of e-books materials.

e-books are also available on CD-ROMs which can be purchased in many book shops and computer shops. Some of these CD-ROMS were compiled through downloads from the Internet.