CHAPTER 3
RESEARCH METHODOLOGY
Somewhere, something incredible is waiting to be known.

~ Carl Sagan

RESEARCH METHODOLOGY

Research is composed of two syllables, a prefix re and a verb search. Re means again, anew, over again whereas search means to examine closely, to test, try and to probe. The two words form a noun to describe a systematic study of knowledge undertaken to establish facts. Research is a diligent investigation into a subject in order to discover or revise facts, theories, applications, etc. Methodology is the system of methods followed particular discipline. Research methodology is the way of conducting the research. Research methodology is a way to systematically solve a problem. In research methodology the various steps that are generally adopted by a researcher in studying the research problem along with the logic behind it is considered. Research methodology may be understood as a science of studying the manner in which scientifically research is performed.

3.1 Achieving Objectives

A combination of theoretical and empirical approach to achieve the objectives is applied. The theoretical approach used relates to the study of component, CBSD, quality factors, quality sub-factors and
identification of metrics. The theoretical approach is based upon review of both secondary data as well as data based on results obtained by applying identified metrics on the obtained component through open source software. Secondary data is acquired from the documenting manuals, research articles, textbooks, journals, technical reports and published thesis. The empirical approach was used for the validation of software metrics comprised of three case studies, for which sources of data collection are source code repositories and software tools. To meet out the analyses a free open source software statistical package PSPP was used. The objectives of the study were achieved by considering nine total versions of three components namely Apache ivy, Heritrix and JfreeChart. To achieve the objective of designing a framework by constructing a structure wherein the attributes of quality structure are defined is accomplished by identifying the quality factors from studying the literature in a systematic manner. The most common and critical attributes of software quality were identified by the study of software quality models. To meet the second objective of identifying the various component quality metrics and to create the framework of these metrics various component quality metrics were identified by analysing various software quality suites to identify the quality metrics related to components. Also various quality sub-factors for the software quality were identified through extensive software engineering literature survey. The third objective of applying the identified metrics on to the component(s) to validate the identified
metrics was achieved by applying two software tools on the components to validate the identified factors. The software tools were Chidamber Kemerer Java Metrics tool and Essential Metrics tool and the components selected were Apache ivy, Heritrix and JfreeChart. To carry out the fourth objective of mapping the developer’s perspective of the various quality factors and sub-factors to the identified metrics the factors were associated to quality sub-factors which were further, correlated to metrics depending upon the validation results performed on applying the software tools on the components. Finally, the objective of proposing a quality assurance model for CBSD was met by proposing a quality assurance framework for CBSD. A relationship between the identified quality factors, sub-factors and software metrics were laid down in the process of building the framework.

3.2 Software Tools Used

Pillar of effective study is selection of suitable research objects and selection of accurate and appropriate software tools. The criteria for selecting research objects and software tools are discussed as follows:

The identified software quality metrics are diversified into two categories, one of the set of metrics related to CK metrics suite and the other set related to MOOD metrics suite. The CK related metrics were WMC, NOC, DIT, RFC and CBO whereas the MOOD related metrics were MHF, AHF, MIF, AIF and PF. The identified metrics are evaluated using software tools. The selection of software tool was based on the following parameters:
1. Availability: The software tool should be available for use preferably from the open source software domain. The ability to execute the tool and look at the source code gives an added advantage.

2. Computes values for identified metrics: The selected software tool should calculate the values of the identified metrics. In case, it is not able to calculate values for all the concerned metrics then at least it should compute for the maximum possible identified metrics.

3. Reliable: The values generated by the software tool should be reliable and tested. Software tool with wrong values should be discarded at the first instance.

4. Efficient: The software tool should be able to compute values very quickly even when the numbers of classes in the software are immense.

5. Minimum Requirements: The pre-requirements for the tool to be executed should be kept to the minimum. High end demands of software tool for RAM, processor, or for high capacity memory devices should be disregarded.

6. In use: The software tool selected for evaluation of metrics should be used by other studies. This removes ambiguity and generates faith in the results.

7. Legal restrictions: The software tool should be legally available for use with no restrictions. Licenses should be easily procurable, if required.
8. Language match: The tool should be able to analyse the component developed in the object-oriented programming language.

9. Size: The tool should have the capability to assess the component having realistic and substantial size.

On the basis of the selection parameters the software tools for measuring the value of identified metrics quantitatively were selected and their description is presented. The identified and selected metrics for the experiment included object-oriented metrics as described by CK suite and MOOD suite namely WMC, RFC, NOC, DIT, CBO, AHF, MHF, AIF, MIF and PF. A thorough search for software tools was made using internet, standard search engines, straight-forward search ways and references from related work. As a consequence, before selecting the final software tool(s), various software tools were downloaded and tested namely, Analyst4j, CCCC, Check Style, Code Pro Analytix, CMT Java, Cyvis, Dependency Finder, Eclipse Metrics Plug in, Java Coding Standard Checker, JDepend, JHawk, Metrics 1.3.6, OO Meter, OXY Project Metrics, Refactor IT, Resource Standard Metrics, SLOC Count, Semantic Design, Semmle, VizzAnalyzer and many more. However, these tools were not able to compute the values for the selected metrics or the programming language the tool analysed was not the desirable one or the results generated by the tools were ambiguous or the tool was not able to compute metrics values of large size components. A particular tool measuring all metrics is not possible, since metrics differ in scope. Finally, after identifying the tools, two software tools namely Chidamber and Kemerer Java Metrics (CKJM)
and Essential Metrics (EM) were selected and used for calculating CK suite metrics and MOOD metrics respectively as they evaluate the identified metrics and also well-matched the parameters mentioned above.

3.3 Research Objects

The research objects refer to the software components from which the software tool is to be applied for extracting the various software metrics values. It is not possible to analyse all the available components as they are present in abundance. To choose the appropriate research component for fulfilling the research objectives short selection process was followed. Keeping in view the objectives of the research the following criteria were laid for the selection of the components:

1. Source Code Availability: The components available under the umbrella of Open Source Software (OSS) have source code availability. To evaluate the value of the identified metrics, source code availability is prerequisite e.g. value of DIT or NOC or AIF can be calculated only if the source code is available. Thereby the selected components are OSS.

2. Availability of Numerous Versions of the Component: A number of releases of the component should exist. It provides an opportunity to the researcher to analyse more than one version of the component and secondly, it represents that a number of attempts are being made to improve the software, making it more stable, error free and effective.
3. Similar Features: The quality metrics framework identified during the research illustrates that the metrics pertain to the object-oriented features. Thereby the component(s) selected for the purpose of analysis should have the object-oriented features namely reusability, inheritance, polymorphism, classes, abstraction and coupling.

4. Programming Language: The developed components while using the concepts of OOPs should have been developed in Java or C++. To maintain consistency across the components selected for analysis same programming language is selected. The components selected are those developed in Java programming language.

5. Widely Accepted: The components should be popular and in demand. The component on daily basis should be downloaded from various locations in the world. This portrays that the software is accepted by the user worldwide and has worth.

6. Standardization. The component should be able to generate results; it should be able to do what it is supposed to do.

7. Downloadable. The component should be easily downloadable.

8. Diversified Nature: Selected components should be from diverse field areas, so that the results of the study can have a wider scope area.

9. Pluggable: It should allow changing parts of the component and letting the user configure accordingly. Component should easily be attached to other components to form software.

10. Ready to Use: Components should be ready for execution.
By using the selection criteria the following components were selected:

1. JfreeChart
2. Apache Ivy
3. Heritrix

The selected components: belong to OSS family, a number of released versions exist for each of them, are developed using OOPs concepts, are popular and widely used with each of these components being downloaded via internet by a number of users daily and are standardised.

Further, for more authenticity and validation of the results three versions of each of these components were downloaded and used in the research.

### 3.4 Statistical Tools Used

Analyzing information involves examining it in ways that reveal the relationships, patterns, trends, etc. that can be found within it. That may mean subjecting it to statistical operations that can evaluate what kinds of relationships seem to exist among variables and also determines to what level the software tool generated results are reliable.

There are two kinds of data generally used for research although not all evaluations will necessarily include both. Firstly the quantitative data refer to the information that is collected as, or can be translated into, numbers, which can be displayed and analyzed mathematically. Qualitative data are collected as descriptions,
anecdotes, opinions, quotes, interpretations, etc. and are generally either not able to be reduced to numbers, or are considered more valuable or informative if left as narratives. Both, quantitative and qualitative information needs to be analyzed differently to extract vital information from the data collected for which statistical tool was applied on the results of experiment. Statistical analysis provides the various methods of analysing the raw data from different viewpoints. The statistical tool selected for the analysis was based on the following considerations:

1. Accurate: The statistical package selected for analysis should compute correct results.

2. Import Data: The data providing mechanism to the package should be simple and easy e.g. the data may be imported from spreadsheets, text files or data base sources.

3. Results: The information in the form of results should be provided in easy format e.g. should have facility to export in HTML and text formats.

4. Interfaces: The user should be provided with choice of terminal or graphical user interface.

5. Compatible: The package should be effortlessly configurable with other software.

6. Popular: The statistical package should be widely accepted.

7. Fast Procedures: The package should compute results fast even on very large data sets.
8. Platform Independence: The tool should be able to run on different hardware and as well as different operating systems.

9. License: Permission to use the package in the form of license should be available. An open source software tool is easily accessible.

Keeping in view the selection considerations Public Social Private Partnership (PSPP) statistical software tool was selected [193].

For the purpose of statistical analysis correlation was performed on the generated results. Correlation is to convert each variable to standard units. The average of the products gives the correlation coefficient. The correlation between two variables reflects the degree to which the variables are related. If there, is a strong association between two variables, then knowing one of them helps a lot in predicting the value of the other. But when there is a weak association, information about one variable does not help much in guessing the other.

<table>
<thead>
<tr>
<th>Value of r</th>
<th>Qualitative Description of the Strength</th>
</tr>
</thead>
<tbody>
<tr>
<td>-1</td>
<td>perfect negative</td>
</tr>
<tr>
<td>(-1 to -0.75)</td>
<td>strong negative</td>
</tr>
<tr>
<td>(-0.75 to -0.5)</td>
<td>moderate negative</td>
</tr>
<tr>
<td>(-0.5 to -0.25)</td>
<td>Weak negative</td>
</tr>
<tr>
<td>(-0.25 to 0.25)</td>
<td>No linear association</td>
</tr>
<tr>
<td>(0.25 to 0.5)</td>
<td>Weak positive</td>
</tr>
<tr>
<td>(0.5 to 0.75)</td>
<td>moderate positive</td>
</tr>
<tr>
<td>(0.75 to 1)</td>
<td>strong positive</td>
</tr>
<tr>
<td>1</td>
<td>perfect positive</td>
</tr>
</tbody>
</table>

Correlation value lies between 1 and -1. A correlation of 1, is referred to as a perfect correlation, there is a perfect linear
relationship between the variables and a positive correlation is that as one value increases the other too increases. A zero value denotes no correlation whereas -1 value means negative correlation meaning that as one value increases, the other value decreases [58].

The most common measure of correlation is the Pearson Product Moment Correlation (called Pearson's correlation for short). Pearson's correlation reflects the degree of linear relationship between two variables ranging from +1 to -1. The Table 3.1 provides an indication of the qualitative description of the strength of the linear relationship and the qualitative value of Pearson correlation (r). The Pearson correlation is a commonly used robust correlation technique.