

# **Final Year Project Proposal**

## **Development Project**



**Project Title: Source Code Metrics Visualizer of Java**

**Project Advisor: Mr. M Ikram Ul Haq**

**Submitted By**

<b>Serial #</b>	<b>Students Name</b>	<b>ID</b>
1	Uzair Zia	17631556-037
2	Ghulam Abbas	17631556-038
3	Farzeen Shahzad	17631556-039

## APPROVAL

I Mr. **M Ikram Ul Haq** am willing to guide these students in all phases of thesis/project titled “**Source Code Metrics Visualizer of Java**” as advisor. I have carefully seen the title and description of the thesis/project and believe that it is of an appropriate difficulty level for the number of students named above.

\_\_\_\_\_  
Date

\_\_\_\_\_  
Signature of Advisor

Submission Date: \_\_\_\_\_

## **Table of Contents:**

<b>Table of Contents:</b> .....	iii
<b>List of Tables:</b> .....	iv
<b>List of Figures:</b> .....	iv
<b>List of Abbreviations</b> .....	iv
<b>1.1 Introduction:</b> .....	1
<b>1.2 Project Title:</b> .....	1
<b>1.3 Background and Existing System:</b> .....	1
<b>1.4 Problem in existing system:</b> .....	2
<b>1.5 Proposed Methodology:</b> .....	2
<b>1.6 Proposed System:</b> .....	3
<b>1.7 Project Overview statement:</b> .....	3
<b>1.8 Project Goals:</b> .....	5
<b>1.9 project Objectives:</b> .....	6
<b>1.10 High-level system components:</b> .....	6
<b>1.11 List of optional functional units:</b> .....	6
<b>1.12 Exclusions:</b> .....	7
<b>1.13 Application Architecture:</b> .....	7
<b>1.14 Gantt chart:</b> .....	7
<b>1.15 Hardware and Software Specification:</b> .....	8
<b>1.15.1 Hardware Components:</b> .....	8
<b>1.15.2 Software Requirement:</b> .....	8
<b>1.16 Tools and Technology:</b> .....	8
<b>1.17 References:</b> .....	9

## List of Tables:

Table 1:	<i>list of Abbreviations</i>	.....	<i>iv</i>
Table 2:	<i>Summary of the Project</i>	.....	4

## List of Figures:

<i>Figure 1:</i>	<i>Flow of Activities</i>	.....	7
<i>Figure 2:</i>	<i>Gantt Chart</i>	.....	8

## List of Abbreviations

Words	Abbreviation
CyVis	Cyclomatic Complexity Visualizer
SCMV	Source Code Metrics Visualizer
2D	Two Dimensional
3D	Three Dimensional
DBMS	Database Management System

## **Chapter 1: Proposal**

### **1.1 Introduction:**

Source code metrics visualization refers to the ability to scan a set of source code and map metrics in graphical forms. The Source Code Metrics Visualizer of java offers many capabilities to visualize an application code metrics.

This way, code flaws and code patterns become more apparent. Such perspectives aim to enhance your understanding of the code base, to let you take better decision to increase code quality and code maintainability.

### **1.2 Project Title:**

Source Code Metrics Visualizer of Java

### **1.3 Background and Existing System:**

Visualization is the graphical presentation of information, with the goal of providing the viewer with a qualitative understanding of the information contents. It is also the process of transforming objects, concepts, and numbers into a form that is visible to the human eyes.

Data visualization is all about understanding ratios and relationships among numbers. Not about understanding individual numbers, but about understanding the patterns, trends, and relationships that exist in groups of numbers[1].

Software visualization is the use of computer graphics and animation to help illustrate and present computer programs, processes, and algorithms. Software visualization systems can be used in teaching to help students understand how algorithms work, and they can be used in program development as a way to help programmers understand their code better[2].

Cyclomatic Complexity Visualizer (CyVis) is a java-based software tool that is used for metrics collection, analyze and visualization. It collects raw data from java class or java files and certain metrics like number of methods, lines, classes, statements, and packages are obtained. When the metrics are collected then they are visualized using visualization techniques[3].

McCabe refers to some typical software analysis metrics for static structure as being, lines of code, number of functions or classes, or complexity regarding graphtheoretical measures, such as cyclomatic complexity[4].

Varieties of JAVA source files are accepted by SCMV, which analyze them to produce object-oriented information. It begins by analyzing every single file to extract class information like attributes, methods and information like data types. Additionally, SCMV provides several types of object-oriented software metrics for each class like number of child and parent class. It further calculates metrics like number of the executable lines, declarations and comments. In order to reduce user's cerebral load, the output of the analysis is visualized in 2 dimensional and 3 dimensional graphs, charts and also generate, print, view and save a report of that information.

To understand the source code for non-programmers was very hectic job, so we visualize source code in 2 dimensional and 3 dimensional charts.

#### **1.4 Problem in existing system:**

Cyvis cannot count data types, blank lines, commented lines, physical line, logical lines, inheritance of class and also not provide 3-dimensional visualization techniques. It cannot facilitate the user to generate report.

To understand the source code for non-programmers was still very hectic job, so we count the level of cyclomatic complexity and visualize source code metrics in 2D and 3D charts and graphs.

#### **1.5 Proposed Methodology:**

Types of Methodology

- Basic vs. Applied
- Qualitative vs. Quantitative

Our methodology which we use in our prototype is

- Applied
- Qualitative

Our proposed research methodology is Applied and Qualitative. Applied because through this product solve the problem facing a non-programmer or even sometime a programmer. Qualitative because we explore the problem and gave the ideas to develop desktop application that has functionality to overcome the problem facing a non-programmer as well as programmers.

**Qualitative Research** is primarily exploratory research. It is used to gain an understanding of underlying reasons, opinions, and motivations. It provides insights into the problem or helps to develop ideas or hypotheses for potential quantitative research.

**Applied research** aims at finding a solution for an immediate problem facing a society or an industrial/business organization.

## 1.6 Proposed System:

The proposed system will perform the following functionalities:

- Browse Java package
- Browse. Java file
- Choose. Java file
- Read file line by line
- Count Metrics
- Establish connection with Database
- Insert counted metrics into Database
- Extract metrics from Database
- Apply visualization Techniques
- Generate Report
- Print Report
- View Report in PDF
- Save Report in PDF

## 1.7 Project Overview statement:

Source code metrics visualization refers to the ability to scan set of source code and map metrics in graphical forms. The Source Code Metrics Visualizer of java offers many capabilities to visualize an application code metrics.

This way, code flaws and code patterns become more apparent. Such perspectives aim to enhance your understanding of the code base, to let you take better decision to increase code quality and code maintainability.

**Table 1: Summary of the Project**

Project Title: Source Code Metrics Visualizer of Java			
Project Manager: Farzeen Shahzad			
Project Members:			
Name	Registration #	Email Address	Signature
Uzair Zia	17631556-037	<a href="mailto:uzairzia143@gmail.com">uzairzia143@gmail.com</a>	
Ghulam Abbas	17631556-038	<a href="mailto:ghulam.abbas4065@gmail.com">ghulam.abbas4065@gmail.com</a>	
Farzeen Shahzad	17631556-039	<a href="mailto:farzeenshahzad18@gmail.com">farzeenshahzad18@gmail.com</a>	

Project Goal: <ul style="list-style-type: none"> <li>• It is useful for new and non-programmers.</li> <li>• It helps users to understand the code in better way.</li> <li>• It is more reliable and efficient.</li> <li>• User friendly interface.</li> </ul>																					
Objectives: <table border="1"> <tr> <th>Sr.#</th><th></th></tr> <tr> <td>1</td><td>Ensures the security of the user's personal information.</td></tr> <tr> <td>2</td><td>It reads the individual java file.</td></tr> <tr> <td>3</td><td>It also reads the Java packages.</td></tr> <tr> <td>4</td><td>It taking out source code metrics and insert into database.</td></tr> <tr> <td>5</td><td>Then it retrieves these stored metrics and visualize in 2 dimensional.</td></tr> <tr> <td>6</td><td>It also provide 3 dimensional visualization.</td></tr> <tr> <td>7</td><td>It generates report in PDF format.</td></tr> <tr> <td>8</td><td>It also facilitates users to view and save.</td></tr> <tr> <td>9</td><td>It also facilitates print the report.</td></tr> </table>		Sr.#		1	Ensures the security of the user's personal information.	2	It reads the individual java file.	3	It also reads the Java packages.	4	It taking out source code metrics and insert into database.	5	Then it retrieves these stored metrics and visualize in 2 dimensional.	6	It also provide 3 dimensional visualization.	7	It generates report in PDF format.	8	It also facilitates users to view and save.	9	It also facilitates print the report.
Sr.#																					
1	Ensures the security of the user's personal information.																				
2	It reads the individual java file.																				
3	It also reads the Java packages.																				
4	It taking out source code metrics and insert into database.																				
5	Then it retrieves these stored metrics and visualize in 2 dimensional.																				
6	It also provide 3 dimensional visualization.																				
7	It generates report in PDF format.																				
8	It also facilitates users to view and save.																				
9	It also facilitates print the report.																				
Project Success criteria: <p>Source Code Metrics Visualizer of Java successfully registered user if he/she have no account and also provide login facility. It counts metrics from java package as well as individual java file. Then it efficiently stored all counted metrics into database and also extract metrics from database and visualize them in 2 dimensional and 3 dimensional such as 3D Line chart, 3D pie chart, 2D Waterfall chart, 2D Dual Axis chart. It also efficiently generates, print, view and save report in PDF format in customized view.</p>																					
Assumptions: <ul style="list-style-type: none"> <li>• Electricity Will be available for 24/7</li> <li>• Internet services available all the time</li> <li>• Hardware support</li> </ul>																					



Risk: <ul style="list-style-type: none"> <li>• High operational risk</li> <li>• Loss of data</li> <li>• Technical risk</li> </ul> Obstacles: <ul style="list-style-type: none"> <li>• Time increase due to burden of other courses</li> <li>• Unexpected shutdowns</li> <li>• Managing expectations</li> </ul>	
Organization Address (if any): Govt. Post Graduate College Jhelum	
Type of project:	<input type="checkbox"/> Research <input checked="" type="checkbox"/> Development
Target End users: <ul style="list-style-type: none"> <li>➤ CEO, Managers who will pay according to line of code</li> <li>➤ Refactoring team</li> <li>➤ Reusability team</li> <li>➤ Reverse engineers</li> <li>➤ Developers</li> <li>➤ New Programmers</li> <li>➤ Software Houses</li> </ul>	
Development Technology:	<input checked="" type="checkbox"/> Object Oriented <input type="checkbox"/> Structured
Platform:	<input type="checkbox"/> Web based <input type="checkbox"/> Distributed <input checked="" type="checkbox"/> Desktop based <input type="checkbox"/> Setup Configurations <input type="checkbox"/> Other _____
Approved By: Mr. M. Ikram Ul Haq	
Date:	

## 1.8 Project Goals:

Project goals are as follows:

- It is useful for new and non-programmers.
- It helps users to understand the code in better way.
- It is more reliable and efficient.
- User friendly interface.

## 1.9 project Objectives:

The main objectives of this project are as follows:

- Ensures the security of the user's personal information.
- It reads the individual java file.
- It also reads the Java packages.
- It taking out source code metrics and insert into database.
- Then it retrieves these stored metrics and visualize in 2 dimensional.
- It also provide 3 dimensional visualization.
- It generates report in PDF format.
- It also facilitates users to view and save the report.
- It also facilitates print the report.

## 1.10 High-level system components:

High level system components of SCMV are as follows:

- **Browser:**  
SCMV have functionality to browse java file as well as java package.
- **Analyzer:**  
SCMV have functionality to read source code line by line and analyse the metrics from selected java source file.
- **Metrics Counter:**  
SCMV have functionality to count metrics from individual java file as well as java package.
- **Database Management:**  
SCMV have functionality to establish connection with database and insert source code metrics into database.
- **Extractor:**  
It can extract source code metrics from database.
- **Visualizer:**  
SCMV have functionality to perform different visualization techniques in 2D and 3D graphs and chart.

## 1.11 List of optional functional units:

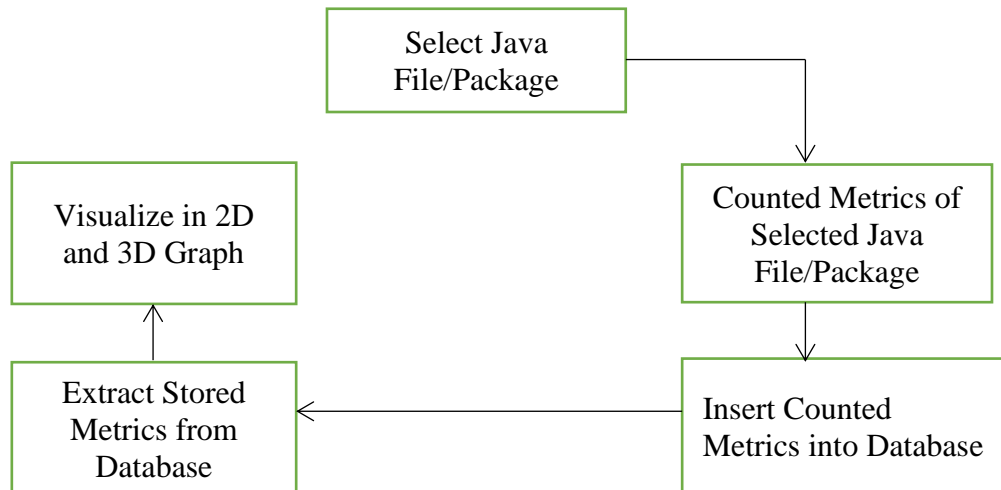
List of optional functional units are as follows:

- This tool is available to target end users at any time whenever they need to visualize their source code.
- This system is easily maintained by the developer or other authorized person.
- User friendly interface

### 1.12 Exclusions:

This study will be beneficial regarding future perspective. As there is a gap in national literature and we will try our best to fill that gap. In this study we will further add more things like we can do refactoring and add more modules and also add visualization techniques in future.

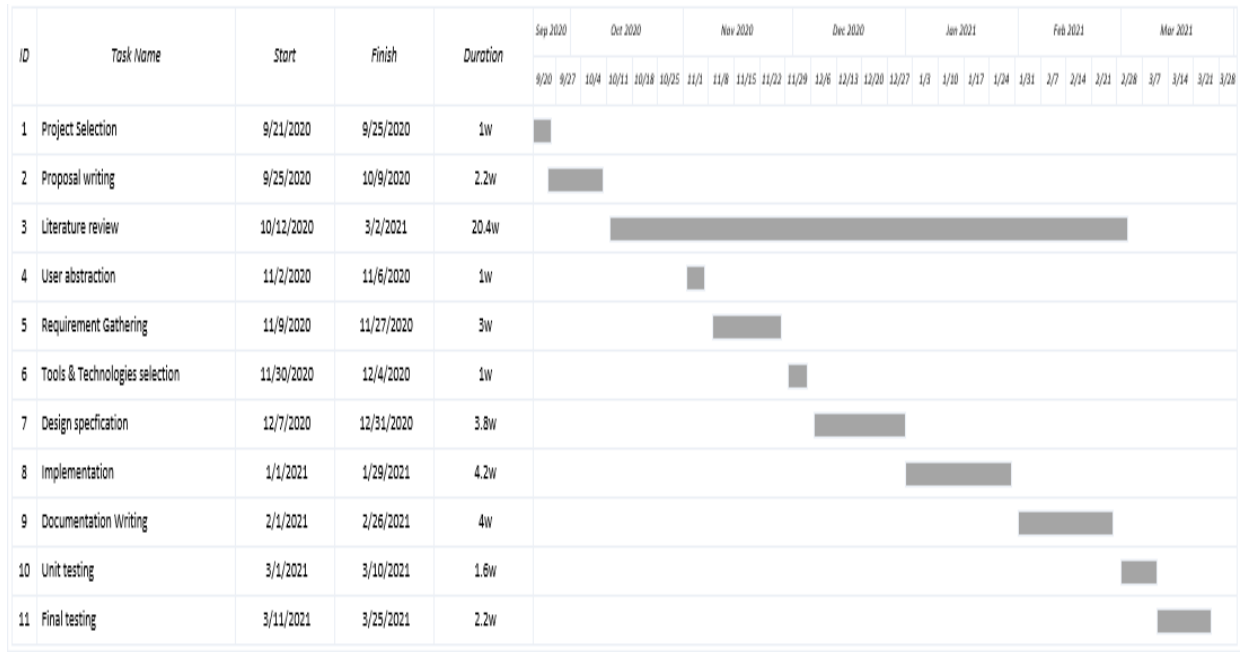
### 1.13 Application Architecture:



*Figure 1: Flow of Activities*

### 1.14 Gantt chart:

A Gantt chart is a timeline that is used as a project management tool to illustrate how the project will run. You can view individual tasks, their durations and the sequencing of these tasks. View the overall timeline of the project and the expected completion date.



**Figure 2: Gantt Chart**

## 1.15 Hardware and Software Specification:

### 1.15.1 Hardware Components:

Hardware components are as follows:

- Processor core 2 duo and above.
- It can support on minimum 50 GB hard disk.
- Memory – 2 GB RAM.

### 1.15.2 Software Requirement:

Software requirements are as follows:

- Windows 7 and latest.
- Net Beans 8.0.2 and above (for Java Programming).
- XAMPP server (for Database Management).
- For UML Design Diagrams:
  - Star UML 5.02
  - Rational Rose
  - MS Visio 2007 and above

## 1.16 Tools and Technology:

The tools and Technology are as follows:

- **Technology:**
  - Java Technology

- **Tools:**
  - **Design Diagrams:**
    - MS office Visio 2007 (for UML 2.0 design diagrams)
    - Star UML v.5.0 (for design diagrams)
  - **Implementation/Coding:**
    - NetBeans 7.4,8.1 or 8.0.2 (for java interface development)
  - **Database Management:**
    - Xampp Server v3.2.2 (for database Management)
  - **Text Editor:**
    - MS word 2010 and latest (for creating documentation)

## 1.17 References:

- [1] Zhao Kaidi, “Data visualization”, School of Computing., National University of Singapore, Iscp0075@nus.edu.sg zhaokaidi@hotmail.com, Matrix Number:HT00-6177E(Document Version 1.0)
- [2] Cc.gatech.edu.Software Visualization research at GVU.2010[online]. Available at: <https://www.cc.gatech.edu/gvu/ii/softvis/>
- [3] CyVis-Software Complexity Visualizer, Cyvis.sourceforge.net,2016[online]. Available at: <https://Cyvis.sourceforge.net/index.html> [Accassed:29-March-2016].
- [4] T. J. McCabe, “A complexity measure,” IEEE Trans. Softw. Eng., vol. 2, no. 4, pp. 308–320, Jul. 1976.