**Ideation Document**

**TeamName:-** Akatsuki

**Team Members:-**

1) Jatin Jha
2) Sakar Jain

**Problem Statement:-** Method Trace Analyzer

**Scope Of Work:-**

Developed GUI using window builder in eclipse It consist of two text fields for writing both passing and failing java class names and a submit button when clicked it will do following tasks:-

• Print name of both class on console
• Call classes which contains functionality of generating log files , comparing them and find the exceptions
• Find number of times each method called and time taken by it , calculate difference in execution of each method for both the classes
• The output is shown on the console and a gui window for graph

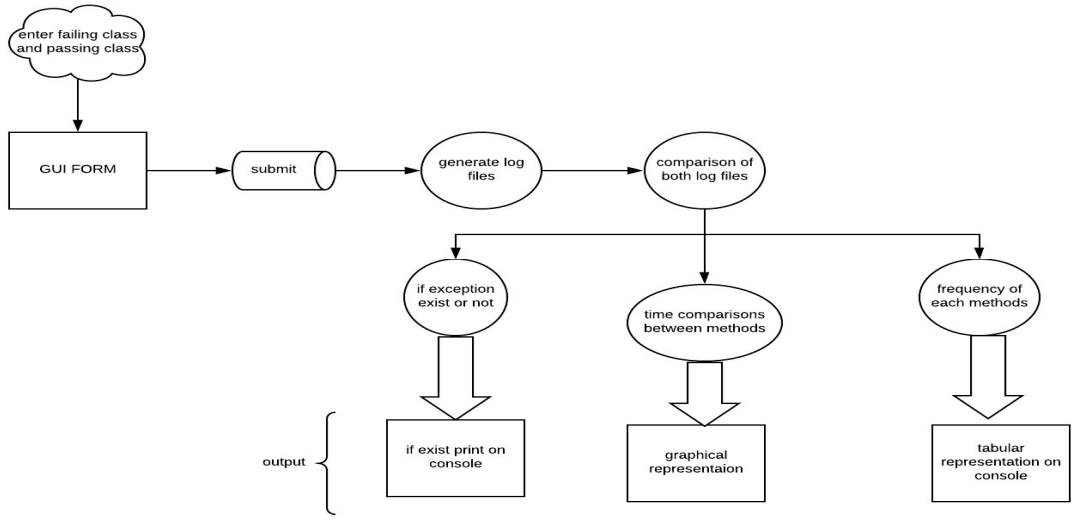
**Solution Impact :-**

Method tracing is one of the commonly used post-mortem diagnostic method to identify problems. It consists of timestamp of entry and exit points for each method invocation. They may also contain stack-trace for each invocation. Depending upon the time for which trace data is collected, the file-size for these traces can be huge. Parsing them manually is a time consuming and error-prone task using this solution post-mortem diagnostic become easy and generated log files can easily analysed which ultimately saves time of a developer

**Technology Used:-**

1) Open JDK with Open J9
2) Eclipse
3) Window Builder
4) JFreeChart library
5) Git

## Architectural flow of the proposed solution




## Work Implemented In Project:-

- Developed java code for generating log files for both passing case and failing case
- Worked on comparing both files using generated log files to find out the anomaly
- Used Xtrace commands to generate trace files and using trace format to convert those unreadable files into
- readable log files
- Developed a java code to read log files line by line and compare them to find out that whether the exception
- exist or not
- Developed GUI using Window Builder where user can input failing case and passing case file names
- Compare methods of both classes to find out which method is taking extra time and represent it using bar
- graph with the help of jfreechart a java library
- represent number of times each method called in tabular form
- Integration of all modules into a single code