

COMPOSITE TEXT ANALYZER

Operational Concept Document

CSE 681 SOFTWARE MODELING AND ANALYSIS

Project# 1

Submitted By: Shrikant Thakare

Term: Fall 2013

Date: 16 September 2013

Contents

1. INTRODUCTION	2
1.1 EXECUTIVE SUMMARY	2
1.2 COMPOSITE TEXT ANALYZER FUNCTIONALITY/ REQUIREMENTS	3
1.3 FEATURES	3
2. POTENTIAL USERS AND USES OF COMPOSITE TEXT ANALYZER	4
2.1 CONTENT PROVIDERS	4
2.2 SOFTWARE APPLICATION DEVELOPERS	4
2.3 SOFTWARE ARCHITECTS	4
2.4 QUALITY ANALYSTS	4
2.5 CODE REVIEW TEAM	5
2.6 PROJECT MANAGER	5
2.7 CUSTOMERS	5
2.8 RELEASE TEAM	5
3. CONTEXT DIAGRAM	5
4. APPLICATION TASKS	6
4.1 PROCESS USER INPUT ON CONSOLE	6
4.2 GET THE REQUIRED FILES	7
4.3 EXECUTE QUERIES	7
4.4 CREATE METADATA	7
4.5 DISPLAY OUTPUT INFORMATION	7
4.6 ERROR HANDLING	7
4.7 ACTIVITY DIAGRAM	8
5. PACKAGE/ MODULE STRUCTURE	10
6. COMPOSITE TEXT ANALYZER MODULE DESCRIPTION	11
6.1 EXECUTIVE PACKAGE	11
6.2 COMMAND LINE PROCESSOR PACKAGE	11
6.3 QUERY PROCESSOR PACKAGE	11
6.4 TEXT SEARCH PACKAGE	12
6.5 METADATA SEARCH PACKAGE	12
6.6 FILE MANAGER PACKAGE	12
6.7 XML PROCESSOR PACKAGE	12
6.8 DISPLAY PACKAGE	13
6.9 METADATA PROCESSOR PACKAGE	13
7. CRITICAL ISSUES	13
7.1 PERFORMANCE OF TOOL WITH LARGE WORKING SET OF FILES	13
7.2 CONFLICTS WHEN CONCURRENT USERS WANT TO ACCESS SAME FILE	14
7.3 MAPPING LARGE SET OF O/P DATA INTO INFORMATION USEFUL TO USER	14
7.4 WORKING OF TOOL WITH FILE TYPES OTHER THAN .TXT, .CS, .DAT	15
7.5 USER CAN INPUT SOME JUNK VALUES	15
8. APPENDIX	15
8.1 SAMPLE OUTPUT OF RESULTS AFTER ANALYSIS	15

1. INTRODUCTION

1.1 EXECUTIVE SUMMARY

In any industry level projects' the number of files can range from 1000 to millions. In such large scale projects it becomes an overhead to manage the files. A tool which will allow to create metadata for each file and which can help to search files based on keyword or random text is a need for software professionals. Composite text analyzer is a tool which intends to do the same.

Composite Text Analyzer is a tool which can perform search operation on the files and displays list of files which contain specific text/ keyword provided by the user. The tool is also capable of creating metadata for a file. This tool will be of great help not only in software industry but also in other industries where the users deal with huge set of files.

It has ten different modules/packages which form the core functionality of Composite Text Analyzer.

Composite text analyzer is a tool which can be used in all phases of a project life cycle.

It is intended to be used by:

- Content Providers
- Software Application Developers
- Software Architects
- Quality Analysts
- Code Review Team
- Project Manager
- Customers
- Release Team

Below four Critical Issues are discussed for the project at the end.

- Performance of the tool with large working set of files.
- Conflicts which can occur when concurrent users want to access same file.
- How well large sets of output data can be mapped into information which will be useful to user.
- Working of the tool with file types other than .txt, .cs, .dat.
- User can input some junk values

1.2 COMPOSITE TEXT ANALYZER FUNCTIONALITY/ REQUIREMENTS

Composite Text Analyzer is a tool which allows user to perform search queries to search occurrence of some text or keyword in set of files. The tool will also give user ability to create metadata for any file specified by the user. Metadata will consist of an xml file which will store details as name, version number, keywords, description, and dependencies for a particular file. The search queries will return the below output:

- Fully qualified names of each file which contains the text specified by the user.

Users can also perform metadata search query on a particular file set. The input will be metadata tags whose information is required. The metadata query will return the below output:

- Content of the tag elements for every file in the file set. If a file has no metadata associated with it, corresponding error message will be displayed.

1.3 FEATURES

Composite text analyzer provides following features to its users.

1. Capable of reading .cs, .txt, .dat files in the specified file set and performs recursive/ non-recursive search to detect occurrence of particular text in the files.
2. Capable of reading xml metadata files and displaying the value of tags requested by the user.
3. Can be executed from Console by providing necessary inputs.
4. Results will be displayed to the user on the Console.
5. The project comprises of eight different packages namely (Executive, Command Line Processor, Query Processor, Display, File Manager, Text Search, Metadata Search and XML Processor). Each package is dedicated to a particular functionality. All packages are discussed in the Package Structure section.
6. Provides tool which can be used to create corresponding metadata file for any .cs, .txt, .dat file specified by the user.
7. Tool to create metadata accepts below inputs:
 - Location of file set.
 - File whose metadata needs to be created.
 - Details as description, keywords, dependencies.

2. POTENTIAL USERS AND USES OF COMPOSITE TEXT ANALYZER

2.1 CONTENT PROVIDERS

Content providers are firm which provides text, graphics of articles based on interviews, developments, stories etc. that can be used to make site or publication useful for readers. They always deal with large set of articles. This tool will definitely be useful to content providers to search for required file using the search by text functionality. Users can enter text which is required to be searched, file set, file patterns and they will be provided with list of files along with full path which has the occurrence of text in it.

2.2 SOFTWARE APPLICATION DEVELOPERS

Developers in a software project work with large number of source code files. This tool can be used by developers to create metadata for the source code files. The metadata file has version number which will help developers for versioning of the files. Using specific keyword/text they can search the required file from large working set. Any application has source code which is branched in packages. The recursive text search functionality of the tool can be used by developers to search occurrence of text through sub-packages as well. This tool will provide developers details about the dependencies on file. They can fetch the all the required files and the files they are dependent on to form a project and start working on it.

2.3 SOFTWARE ARCHITECTS

Software architects are engaged in working on multiple projects at the same time. Architects can use this tool to create metadata for a file to associate the file with particular project. The metadata tags can be used to associate the file with project. E.g. Keyword tag can be used to enter project name which file is associated with. Also they can use metadata search functionality to view metadata for files from work set. Architects will be able to get the required files and their dependencies which will help them to study a project. They can also maintain different versions of file and get any of the versions whenever required.

2.4 QUALITY ANALYSTS

QA can use the tool to get different versions of a file. QA can download source code and also the files/ APIs on which it is dependent and then review entire project. QA will get to know about the updates done by the development team and he/ she will be able to track the changes. QA can extensively use search functionality to get the required files.

2.5 CODE REVIEW TEAM

Code review team generally is involved in reviewing the code of the project against set of metrics which are followed in Software industry. Software project consist of thousands of code files, this tool will help code review teams to get the file which needs to be reviewed using text search functionality. They can also use metadata search functionality to get details of tags files with similar keyword. They can get the any version of file which needs review. Also, the review team can track changes done in file for different versions.

2.6 PROJECT MANAGER

Project Manager can use metadata search functionality to get the list of versions of a project management file and then download required version. Manager will also be able to get the details about the user who created a file. He/ She can also track the file versions using the metadata search functionality.

2.7 CUSTOMERS

Customers deal with large number of files. The tool will ease customer's task to manage large set of files. Clients are provided facility to create metadata for a file by specifying different details about t the file. The tool on the backend will maintain details as timestamp of file creation, user who created the file and add it to metadata file.

2.8 RELEASE TEAM

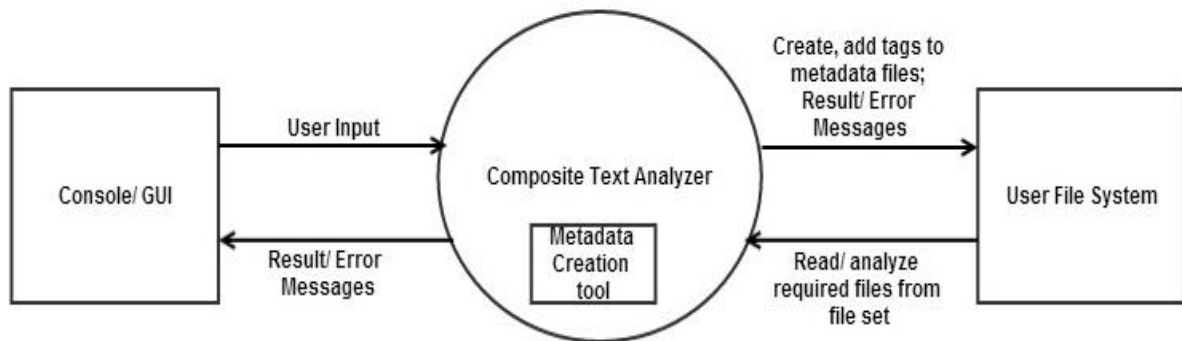
Release team is involved in creating build and deploying the project on production environment. The tool will assist the release team in getting source code files with required versions and also the details about the APIs the code files are dependent on. They can then create build for project using the code files and deploy it. The metadata creation tool will help them to create metadata for different versions of build.

3. CONTEXT DIAGRAM

Context diagram shows how Composite Text Analyzer interacts with the other components within the environment in which it is being executed.

From high level perspective Composite Text Analyzer interacts with the Console/GUI and the File System.

CONTEXT DIAGRAM FOR COMPOSITE TEXT ANALYZER



Interactions of Other Components with Composite Text Analyzer:

- a. **Console/ GUI:** The user gives its input to Composite Text Analyzer and Metadata Creation tool on the Console/ GUI. The input for Text Analyzer is path of the file set, text to be searched or list of tags required by user and switch's options ('/A', '/O'). The Text Analyzer performs text/ metadata search query and returns the results to the Console/ GUI. The input to the Metadata Creation tool is the file name whose metadata needs to be created, details about the files. The creation tool creates metadata file and returns success/ error message to the console.
- b. **File System:** Text Analyzer searches .cs, .txt, .dat files from the File System. For metadata search it searches the .xml files from the File System. The metadata creation tool will create the metadata file on the File System.

4. APPLICATION TASKS

Application tasks summarize the steps performed by Composite Text Analyzer in order to achieve its task. Following are the major steps performed by the tool.

4.1 PROCESS USER INPUT ON CONSOLE

The user enters file pattern, location of file set, text to be searched, switch's as '/A' or '/O' on the console. For metadata search, user will enter file pattern, location of file set, list of

tags on the console. For creating metadata of file, user will enter file name whose metadata needs to be created, details as keywords, description, dependencies etc.

4.2 GET THE REQUIRED FILES

Once user provides location of the file set and file types, a check is done to verify if the file set specified by user exists. If it exists all the files which match the criteria are fetched and a list is created with their absolute paths. If file set do not exist message is displayed to the user. If the search query is executed with recursion (/R) file set will include all files in the directory tree rooted at given path.

4.3 EXECUTE QUERIES

Search query and metadata queries are type of queries which can be executed by the tool. Search query will perform search on all the files in the file set to detect occurrence of text/ list of texts and will create list of those files. If user has specified switch's as '/A' or '/O' it will be added to the query. Metadata query will perform search on all the metadata files and get the values of tags requested by the user. If metadata is not present for file, error message is appended to output.

4.4 CREATE METADATA

User will specify the file whose metadata needs to be created along with the details as keywords, description, dependencies. A check is done to verify whether the file whose metadata needs to be created exists. If it exists, details are used to create corresponding metadata file (.xml). If the file do not exists message is displayed to user.

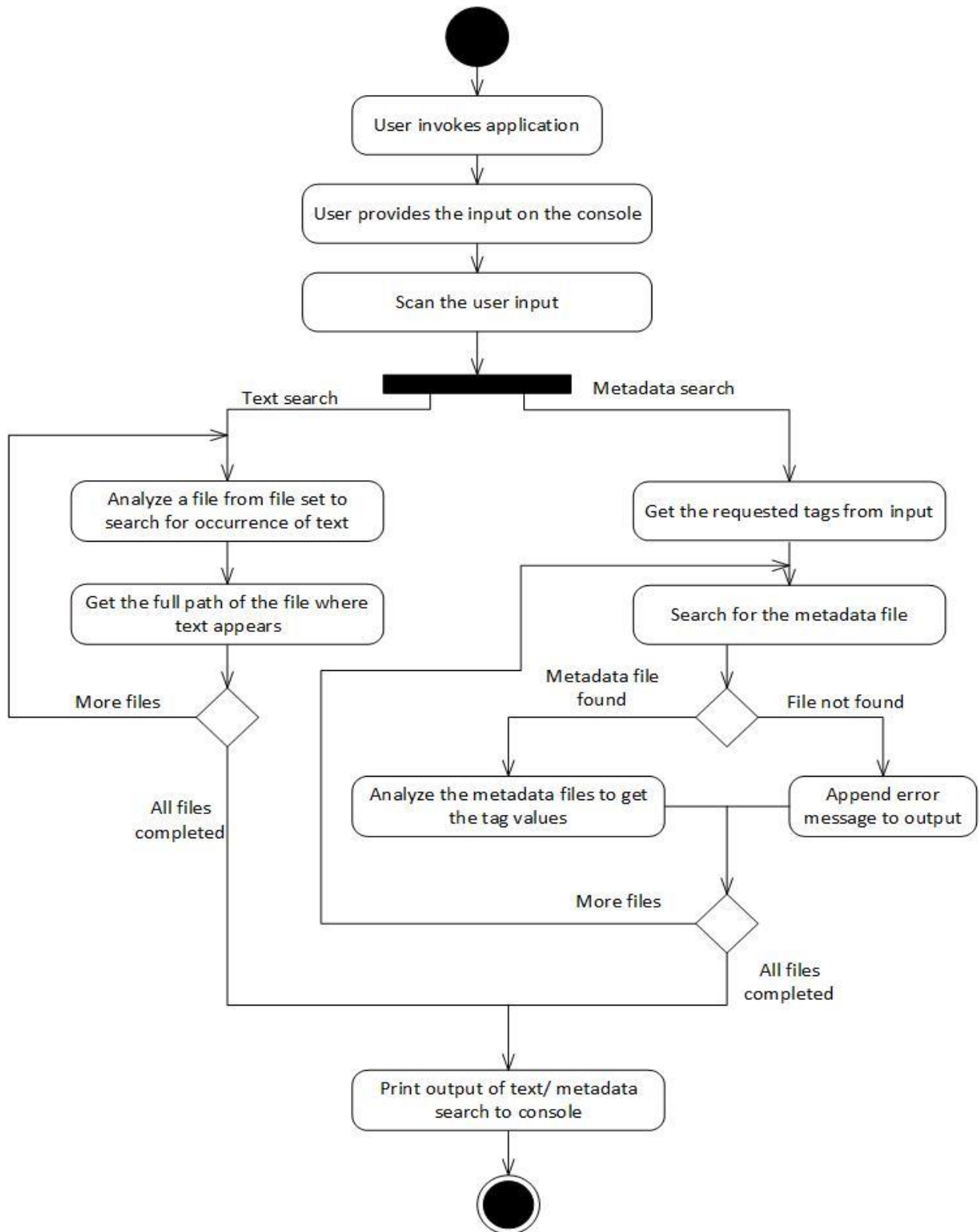
4.5 DISPLAY OUTPUT INFORMATION

The result of search query is directed to console. Text search query will output list of files with fully qualified path on the console. Metadata search query will output values of tag elements requested by the user. The output will have file name and values of tags from file's metadata. For metadata creation tool, the output will be message which will inform user whether the metadata has been created or not.

4.6 ERROR HANDLING

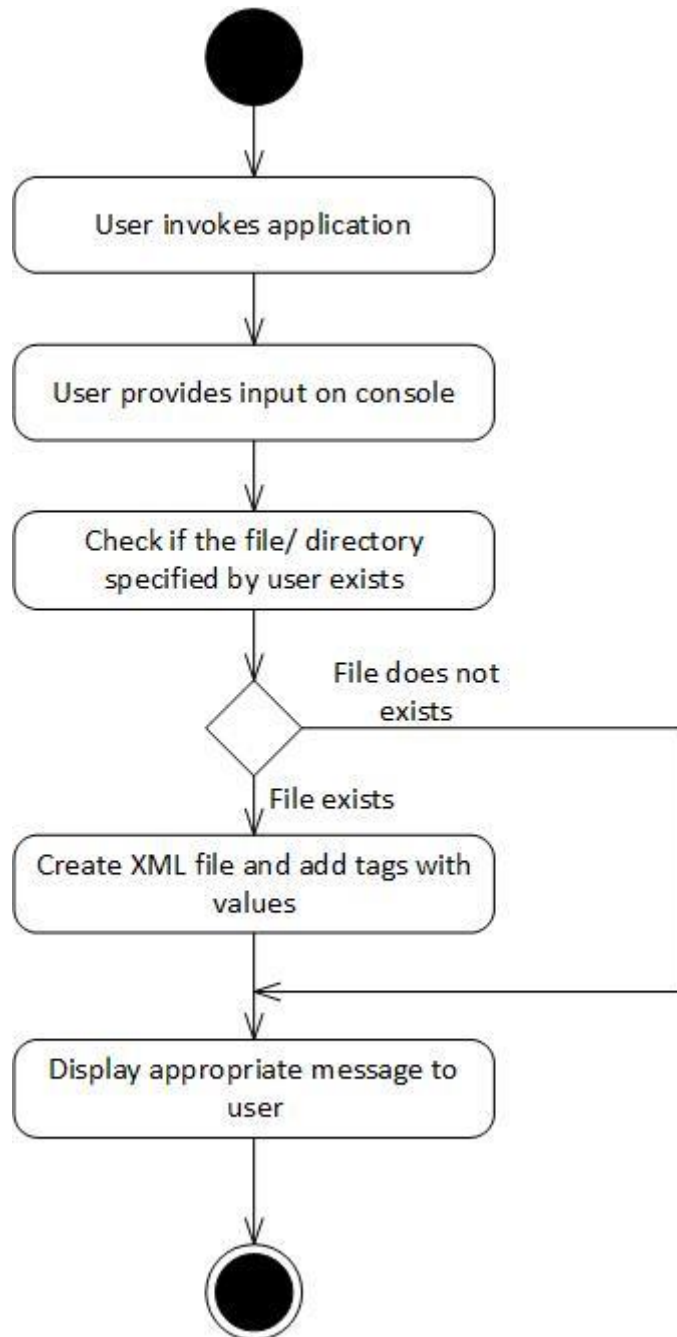
Error handling will be important in the Composite Text Analyzer tool. Proper error message will be displayed to the user at every step where exception will occur during the execution of the tool. Also if input is not entered properly user will be prompted with proper message.

4.7 ACTIVITY DIAGRAM



ACTIVITY DIAGRAM FOR COMPOSITE TEXT ANALYZER

COMPOSITE TEXT ANALYZER

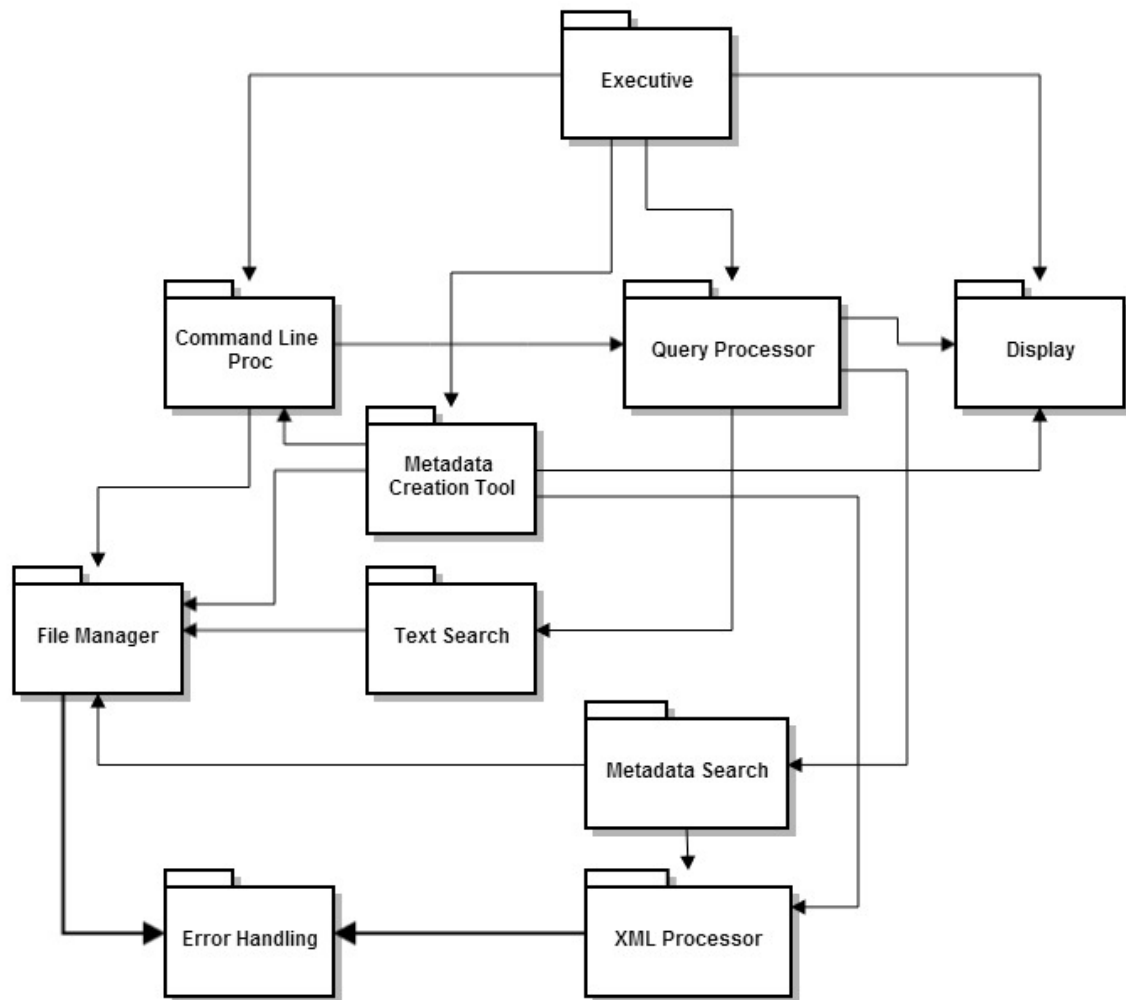


ACTIVITY DIAGRAM FOR METADATA CREATION TOOL

5. PACKAGE/ MODULE STRUCTURE

The following diagram represents the partitioning of the Composite Text Analyzer in different packages. In total there are ten packages with Executive as the main controller package.

Brief description of packages is discussed after the diagram.



Package diagram for Composite Text Analyzer

6. COMPOSITE TEXT ANALYZER MODULE DESCRIPTION

6.1 EXECUTIVE PACKAGE

Executive package is the controller package of the Composite Text Analyzer tool. The main responsibility of this package is to co-ordinate with the other packages of the tool. It also acts as an entry point for the tool. It is responsible for below operations:

- To invoke methods from other packages of the tool depending on the operation requested by the user.
- Provide inputs to the invoked methods.
- Store output of these methods and display it to user using the Display package.
- Use output of one package to instantiate methods of other packages.

6.2 COMMAND LINE PROCESSOR PACKAGE

This package is used to handle the input provided by the user on command line. It accepts the command line parameters provided by the user. It performs following operations:

- It scans the input provided and determines the operation requested by the user.
- It forwards the path of the file set provided by the user to the File Manager.
- It forwards the input query to the Query Processor.

For metadata creation tool it accepts file name for which metadata needs to be created, details as description, keywords, dependencies etc.

6.3 QUERY PROCESSOR PACKAGE

This package process the query forwarded by the Command Line Processor package to determine whether the request is for text search or metadata search. Depending on the request it forwards the request to Text Search or Metadata Search package. Below are the operations the package performs:

- It filters the input query and invokes Text Search or Metadata Search package.
- Sends the output from Text Search/ Metadata Search package to the Display package.

6.4 TEXT SEARCH PACKAGE

This package search for occurrence of particular text in the files which are part of specified file set. It accepts the text or sequence of texts to be searched in the files. Below are the operations the package performs:

- It uses File Manager package to get the files in the file set.
- It performs text search on the files.
- It sends back the list of files which contains the text to Query Processor package.

6.5 METADATA SEARCH PACKAGE

This package scans the metadata files to fetch the details. It accepts the tags whose details are required.

- It uses File Manager package to get the files in the file set.
- It fetches the details from the metadata file if present and sends it to Query Processor.

6.6 FILE MANAGER PACKAGE

This package accepts the file path, file patterns from the Command Line Processor package which acts as file set for the tool. This package uses the capabilities of .Net framework for file handling mechanism.

- It performs recursive or non-recursive search for the files in directory (file set) specified by the user.
- It gets the list of files depending on the extensions (.cs, .txt, .dat) and sends it to Text Search or Metadata Search package.

6.7 XML PROCESSOR PACKAGE

This package is used to perform xml operations on the metadata xml files. It accepts the contents of metadata file, xml tags as input.

- It retrieves the values of required tags from the metadata file and sends it back to Metadata Search package.

For metadata creation tool it accepts tag values from metadata processor and creates metadata file (.xml) in the directory.

6.8 DISPLAY PACKAGE

This package is used to display the result of the search queries on the console. It also displays error message to the user in case if user input is not proper. For metadata creation tool it displays success/ error message to the user.

6.9 METADATA CREATION TOOL PACKAGE

This package accepts metadata details from the command line processor package and adds them to respective tag elements. It then forwards these tags to XML Processor package to create metadata file.

6.10 ERROR HANDLING

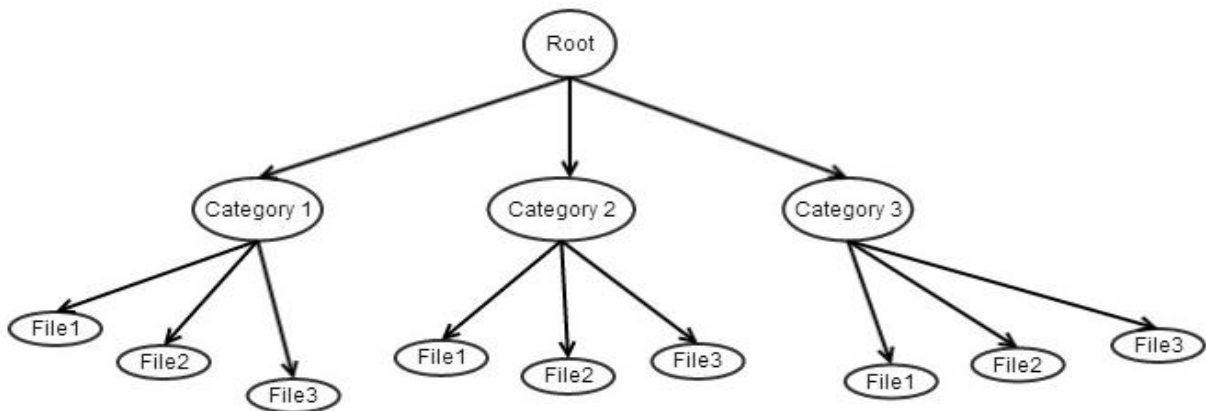
This package is responsible to handle all the possible exceptions which will be thrown by File Manager and XML Processor package. It handles the exception and returns proper error message which user is able to interpret.

7. CRITICAL ISSUES

7.1 PERFORMANCE OF TOOL WITH LARGE WORKING SET OF FILES

The Composite Text Analyzer tool will be used to perform search on large file set. The performance of the tool will depend on the size of the file set provided by the user. The performance will be inversely proportional to the size of file set. As the size of the file set increase the performance of the tool will degrade. Also if user wants to perform recursive search then it will add to bad performance of the tool.

Solution: One of the solutions to this problem can be categorizing the files which are part of file set. Category will be used to filter the files which need to be searched. This will increase the performance of the tool.



7.2 CONFLICTS WHEN CONCURRENT USERS WANT TO ACCESS SAME FILE

There may be situation where more than one member of project team is using the tool and wants to get metadata details. In such cases file will be locked when first user is accessing it so other users will not be able to access it. The tool can throw exception in such cases which user won't be able to know about.

Solution: The solution to this problem is to check whether a file is locked. If file is locked, exception should be caught and proper error message should to be displayed to the user rather than throwing exception. This will help user to better interpret the output.

7.3 MAPPING LARGE SET OF O/P DATA INTO INFORMATION USEFUL TO USER

If user performs text search on a large file set, the output returned will be list of all the files which has occurrence of the text. The output list will go on increasing with the size of file set. For metadata search, the size of output will depend on the tags requested by the user and also on the size of the file set. Thus, it's very important to map the output data into information which user can interpret.

Solution: If the user requested for text search then the list of files which will be displayed to user can be sorted by the directory its contained in and then sorted by the file name. If the user requested for metadata search the output can be framed such that initially file name and its metadata file name is displayed and then the value of tags is displayed. If the metadata file is not present the error message is displayed. This will provide the user with the required information.

7.4 WORKING OF TOOL WITH FILE TYPES OTHER THAN .TXT, .CS, .DAT

The tool can be used to perform operation on files with type as .txt, .cs, .dat. Source code files are mainly .cs files, so developers can use the tool to perform a search. Project Managers mainly deal with project management documents (.doc) files. As of now, the tool can be used only on .txt, .cs, .dat files. This will restrict the use of the tool.

Solution: To get the tool working for .doc files, a reference needs to be made to Microsoft.Office.Interop.Word

7.5 USER CAN INPUT SOME JUNK VALUES

The tool accepts various input parameters from user. In situation where the input entered by user is not proper, tool may crash if it is not handled properly.

Solution: Before processing the query, user input needs to be scanned to verify whether the query entered is proper. User needs to be prompted to enter proper input if it's not proper.

8. APPENDIX

8.1 SAMPLE OUTPUT OF RESULTS AFTER ANALYSIS

Output for text search functionality

```
Text to be searched: threads
List of files where text "threads" occurs
1) C:\Compositetextanalyzer\thread.h
2) C:\Compositetextanalyzer\thread.cpp
```

Output for metadata search functionality

```
File Name: thread.cpp
Metadata file name: thread.xml
Tag Values: Keyword = thread
            Version = 1
```

```
=====
File Name: thread.h
Metadata file name: File does not exists
=====
```