

CHAPTER IV

ANALYSIS AND DESIGN

4.1 Analysis

4.1.1 Use Case Diagram

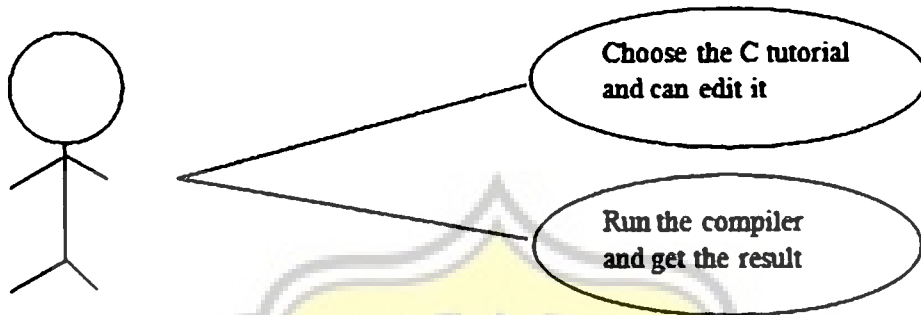


Figure 4.1 Use Case Diagram

4.1.2 Activity Diagram

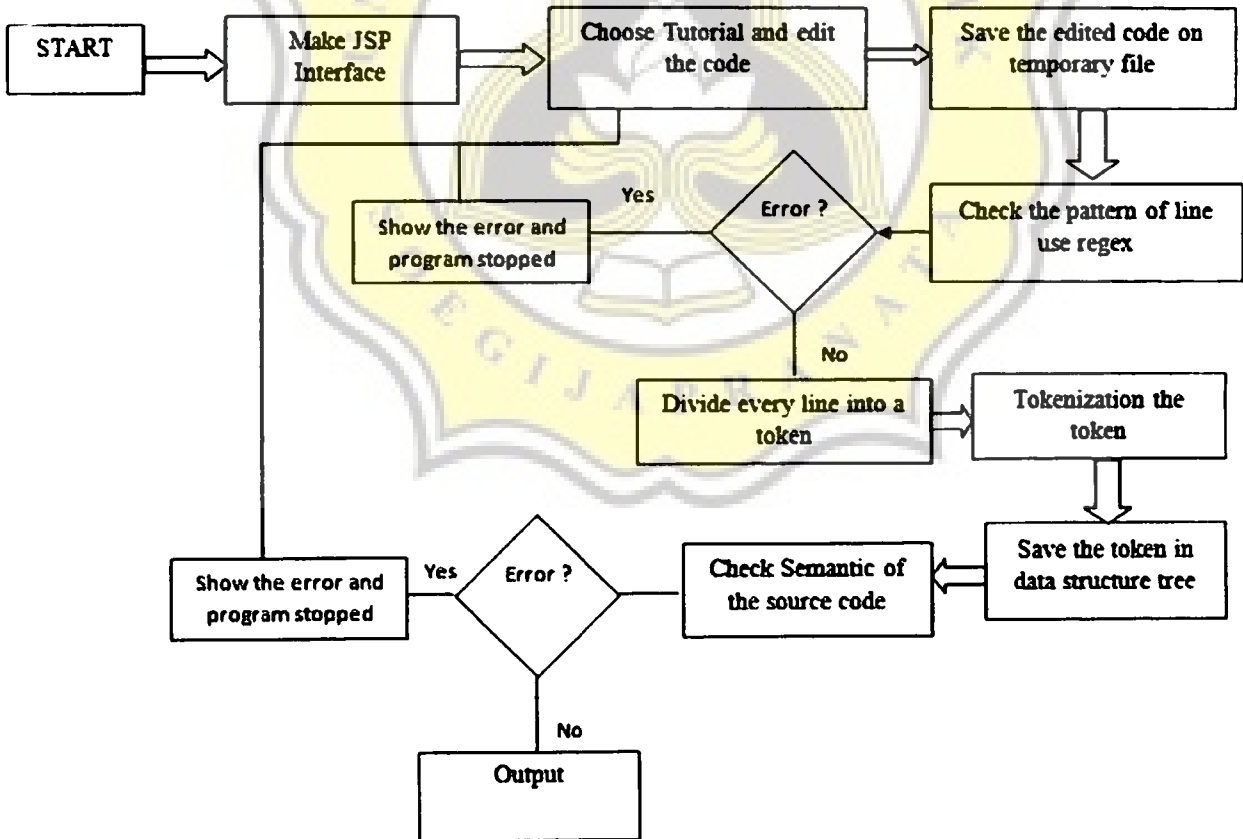


Figure 4.2 Activity Diagram

4.2 Design

4.2.1 Class Diagram

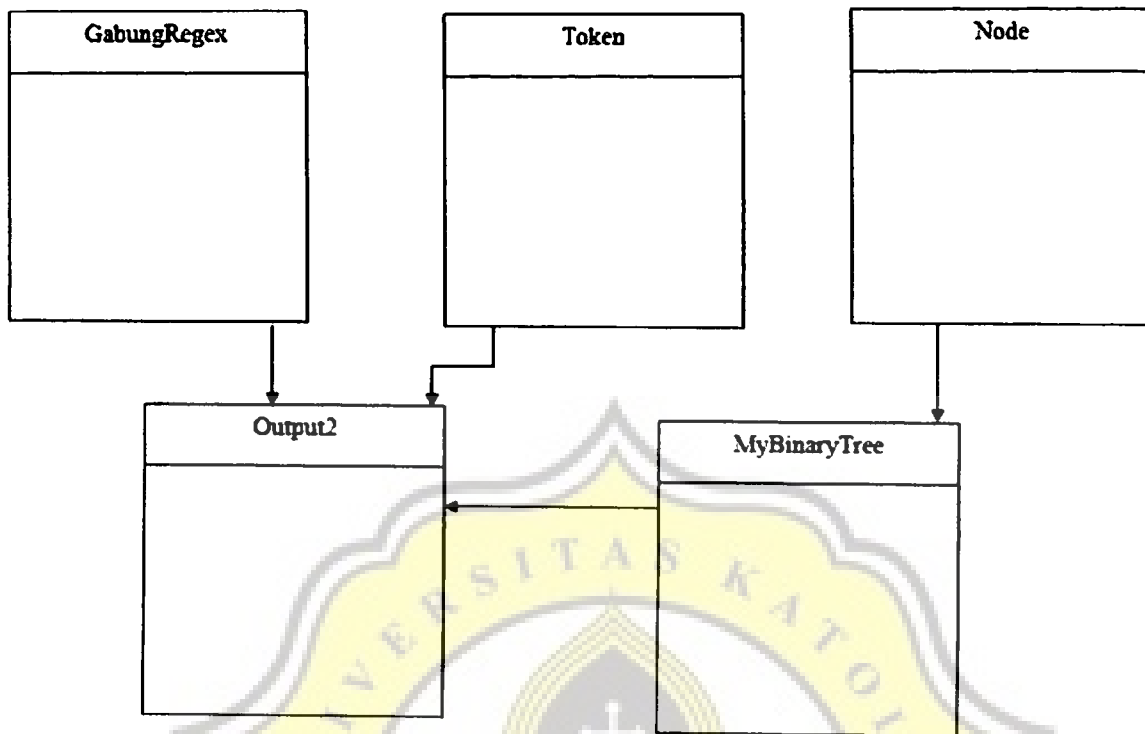


Figure 4.3 Class Diagram

In this class diagram, communication between class started from Output2, which is every input source code processed in here. Output2 will call class GabungRegex to check pattern every line and send the result back into Output2 to be shown the result. After that, Output2 will call class Token to proceed token and tokenization process and send return value to Output2 to show the result of token and tokenization. Output2 send the token that got from class Token to MyBinaryTree. Class MyBinaryTree will call class Node to save root and node of the tree. And send it back to Output2 to show the result. And the semantic process are in Output2 class. The result also shown in Output2 as a JSP.

4.2.2 Class Diagram Detail

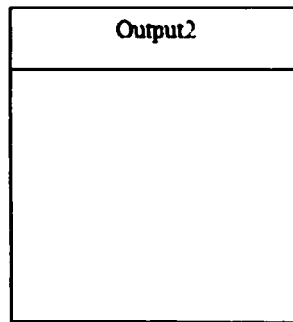


Figure 4.4 Output2 Class

The main purpose of Output2 class is as a JSP interface and receive user input and call classes that needed. Output2 also call all of the class except class Node. In this page result of every step shown.

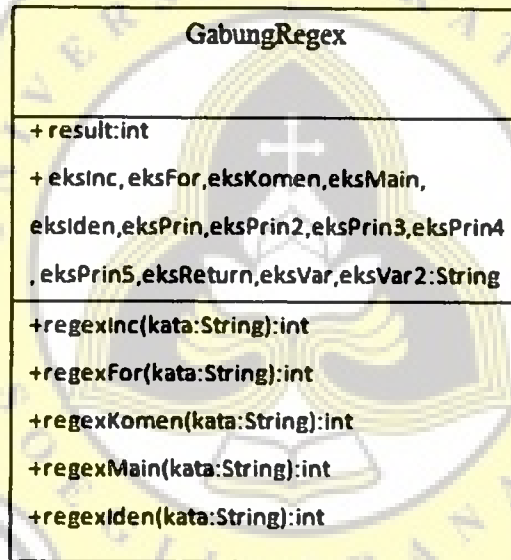


Figure 4.5 GabungRegex Class

The main purpose of class GabungRegex is checking the pattern of source code. Checking process is by each line. In this class will return value 1 if true and 0 if false and send it back to Output2.

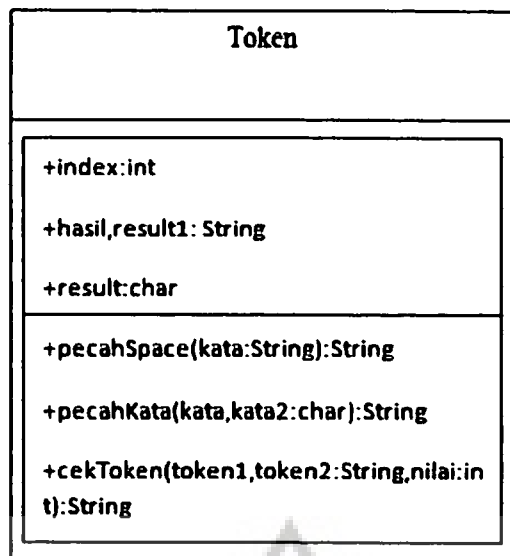


Figure 4.6 Token Class

The main purpose of Token class is to create token of source code. After token created, this class also describe tokenization of each token. This process also do by line per line.

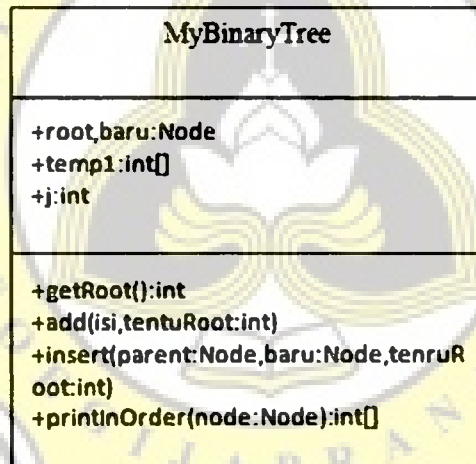


Figure 4.7 MyBinaryTree Class

The main purpose of this MyBinaryClass is to create a tree from input token. After creating tree, this class send the data into Output2. This class also can access Node class, so could get data from class Node.

Node
+isi, i, tentuRoot, temp:int +ki, ka:Node
+Node(isi) +Node() +getIsi():int +getKa():Node +getKi():Node +setIsi(i:int)

Figure 4.8 Node Class

The main purpose of this class is to make node as basic of tree. The tokens that created in Lexical Analysis process also saved in this Node. This class will be called by MyBinaryTree class.

