

CHAPTER IV ANALYSIS AND DESIGN

4.1 Analysis

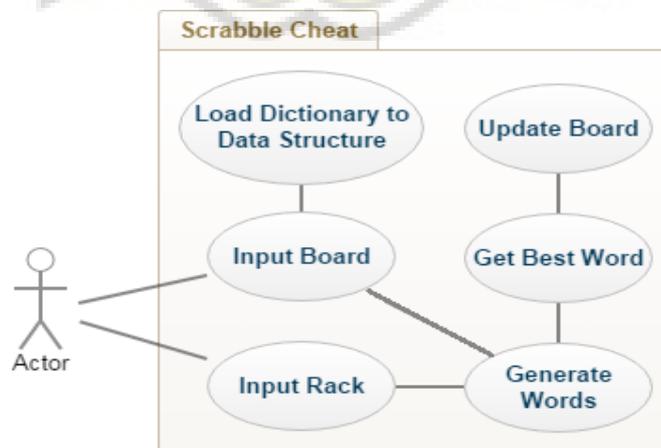
The purpose of this project is to make a cheat for scrabble game. This cheat will help the player to find the words that can be formed and with the highest bonus value. Cheat application needed data such as dictionary. In this project use a dictionary with 1,000 vocabulary. To store vocabulary using Trie data structure. Trie data structure can save many vocabulary efficiently.

The algorithm will be used is GADDAG algorithm. GADDAG algorithm can form a word based on prefix of word. This algorithm will process the data structure that can store all possible word efficiently.

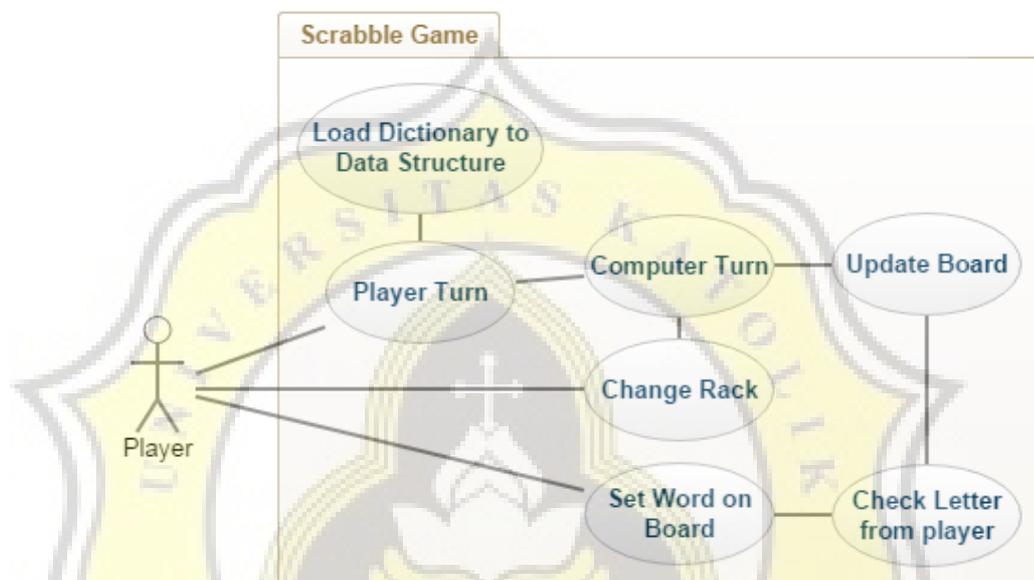
To find words that can be formed need to analysis board and letters are owned by players. GADDAG algorithm searches for any location at board that has a chance to form a word by comparing the dictionary on data structure. After finding all possible words that can be formed, and then the program will choose a word that has the highest bonus points. The word points are calculated based on a Scrabble board in accordance with the Scrabble rules.

4.2 Design

4.2.1 Use Case Diagram

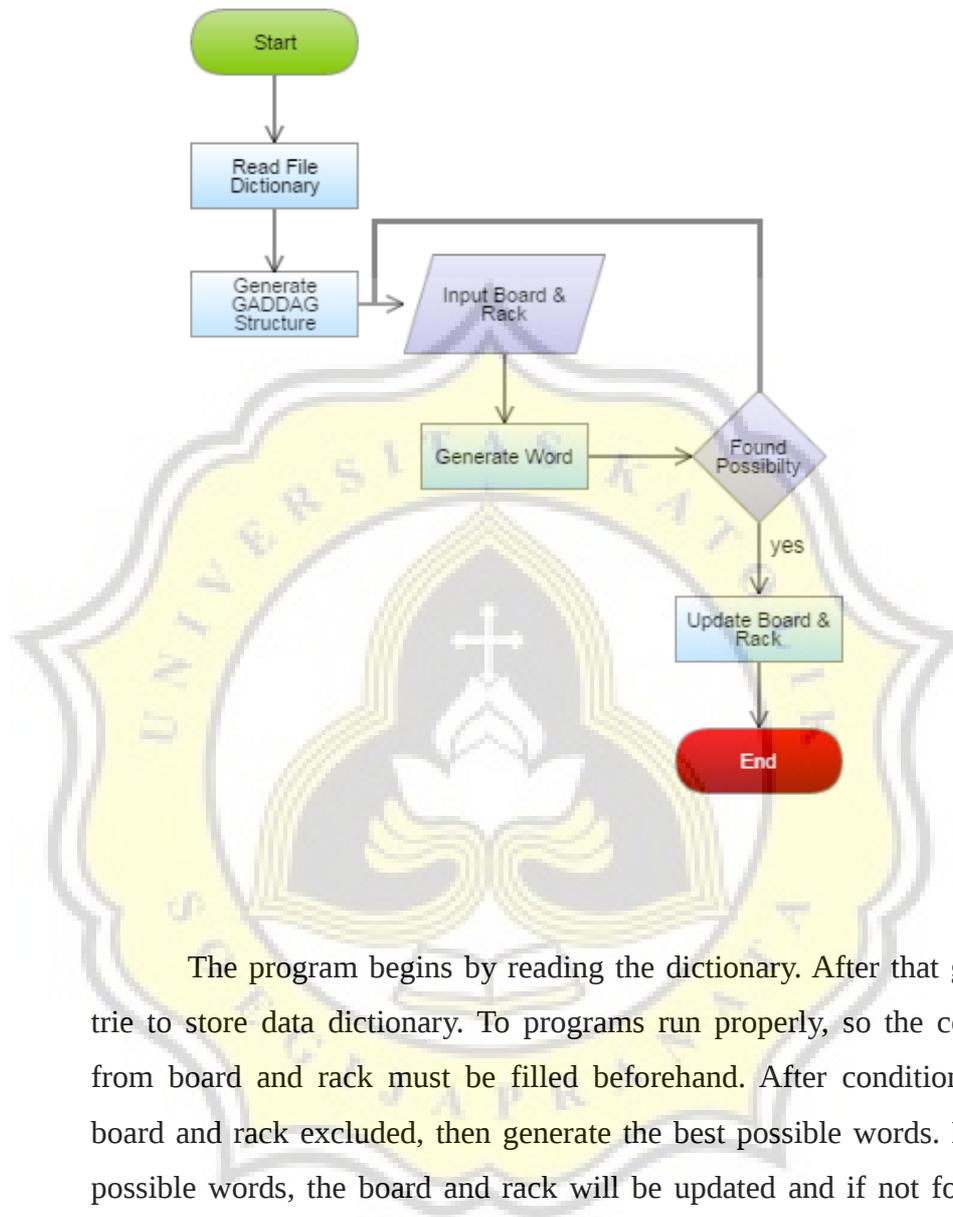


The program start with load dictionary into data structure. After that User input the condition of board. Users can input the condition of the board and the rack (letters owned). After determined the board and rack the system will generate the best words. If found word system will update board and rack condition.



The program start with load dictionary into data structure. In the Scrabble game, player will get first turn to play. Players can have two options, set letters on board or exchanging letters owned. If a player enters a word, the system will check whether a valid word. If the word is valid then system will update point and condition of board and computer turn to play. Otherwise letter on board will return to player rack. Check letter is using the same procedure on cheat scrabble for search all possible word. If the player swapping rack, then computer turn to play. After the computer turn ends then player turn to play.

4.2.2 Flow Chart



The program begins by reading the dictionary. After that generate trie to store data dictionary. To programs run properly, so the condition from board and rack must be filled beforehand. After condition of the board and rack excluded, then generate the best possible words. If found possible words, the board and rack will be updated and if not found, go back to the initial conditions.

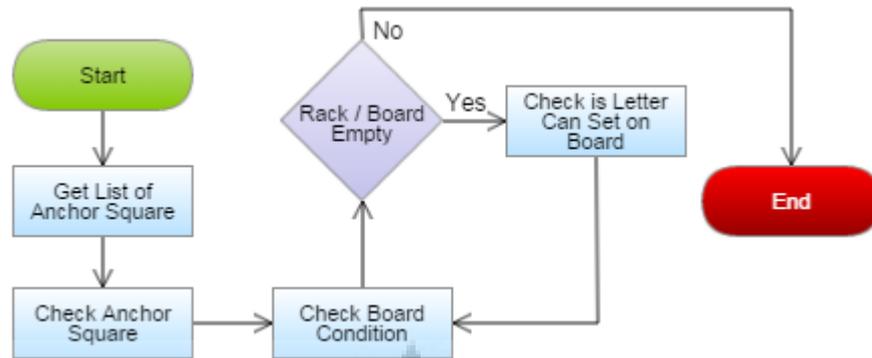
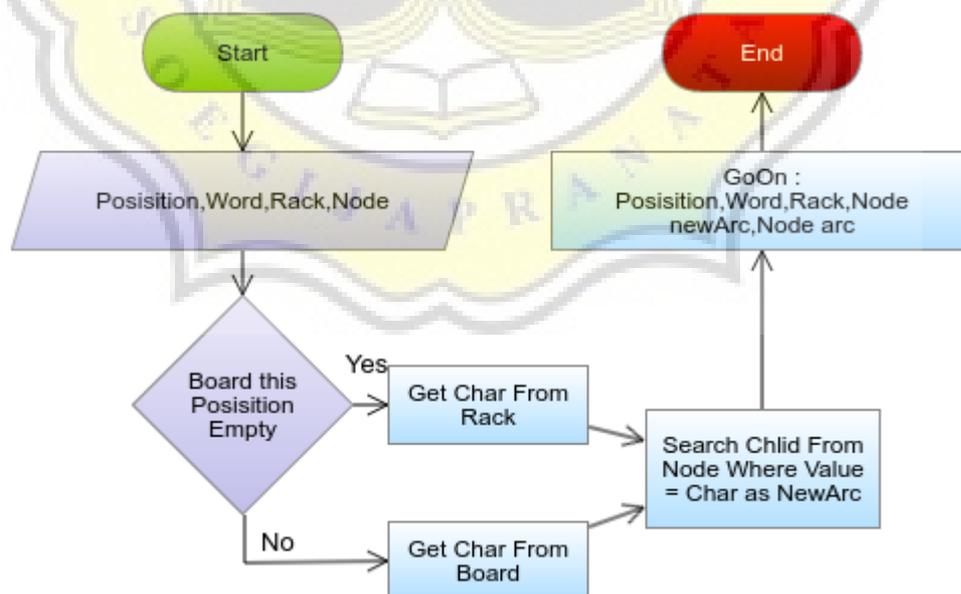


Figure 8: Flowchart Generate Word

The search process for the best possible starts with finding Anchor Square first. Anchor Square is right next to the coordinates of the letters on the board. After that starts checks every Anchor Square to be filled letter from the rack (letters owned). The search results will store in a list. After that the program will choose the word with the highest score. In GADDAG algorithms have procedures to search steps, check condition board(Gen) and check is letter can set on board(Goon). Both of these procedures run recursively.



Check Condition Board procedure called Gen procedure on the algorithm GADDAG. This procedure has a role to take each letter of the rack and then matched to the appropriate position on the board and with the dictionary. Position is used to movement position on board. In first time start position always get value -1.

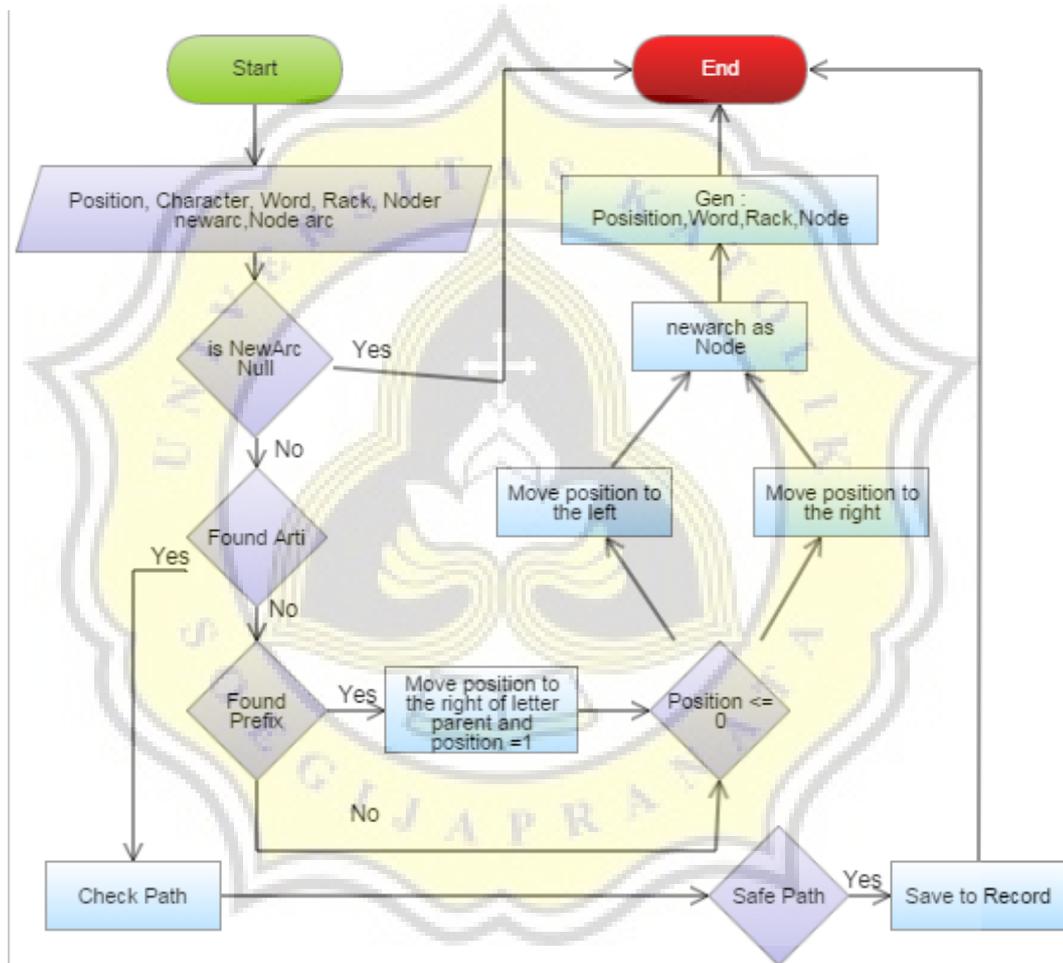
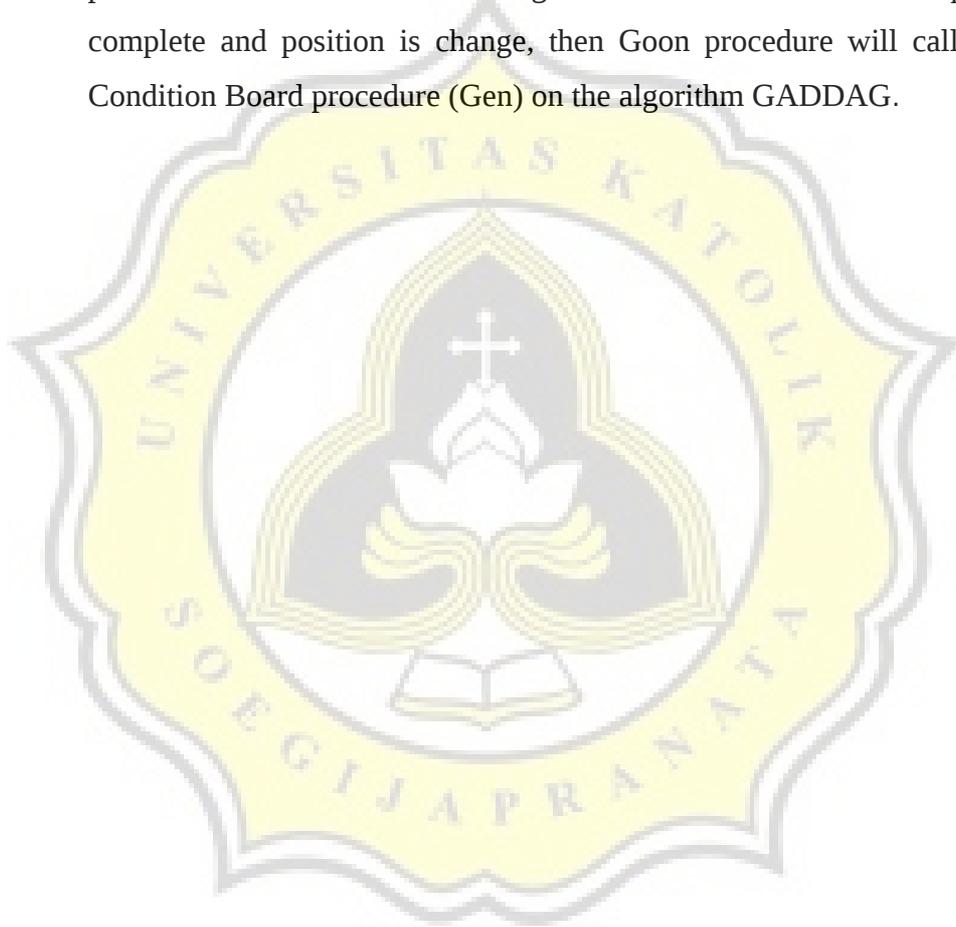


Figure 10: Flowchart Check is Letter Can Set on Board (Goon)

Check is Letter Can Set on Board (Goon) procedure on the GADDAG algorithm, this procedure has a role to check whether a letter could be paired on the board. If a letter can be fitted and find meaning it will be added to the record. If it finds the prefix (>) position will be shifted to the left (position -1) will instead be shifted to the right (position +1). Position right on Anchor Square have a value 0 and if position to the right position will be -1 and if to the right will be +1 and so on. After process, complete and position is change, then Goon procedure will call Check Condition Board procedure (Gen) on the algorithm GADDAG.



4.2.3 Class Diagram

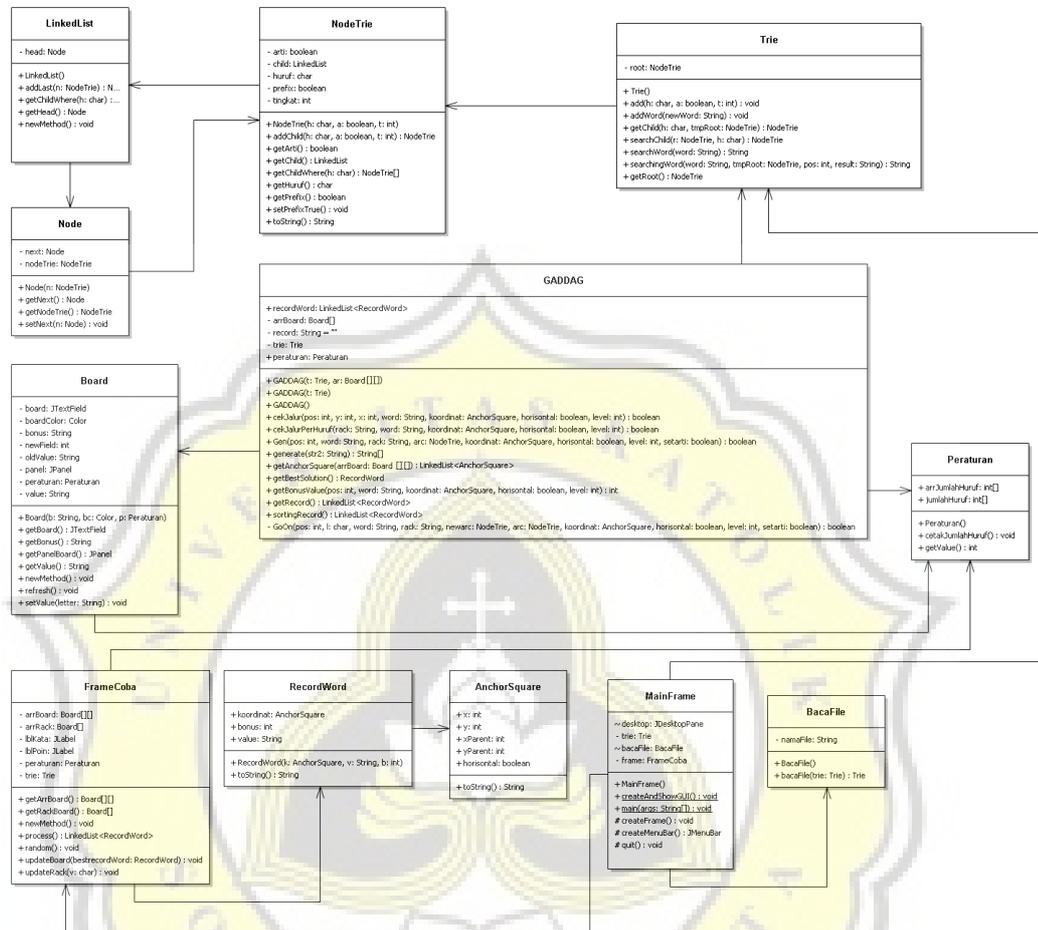


Figure 11: Class Diagram

In the above design is divided into three categories there are, data structure, algorithm GADDAG, save the record of word and board coordinate, and input output data. GADDAG class is the main to search move generation word from scrabble.