Implementing Word Search Game With Dynamic Grid Generation And Analyzing Its Benefits

Tanmayi Kulkarni[1], Maitreyee Shendye[2], Yuga Tungar[3], Prof. Nutan Deshmukh[4]
[4]Assistant Professor, Computer Engineering Department,
MKSSS's Cummins College of Engineering for Women, Karve Nagar, Pune, India.
[1]tanmayi.kulkarni@cumminscollege.in, [2]maitreyee.shendye@cumminscollege.in,
[3]yuga.tungar@cumminscollege.in ,[4]nutan.deshmukh@cumminscollege.in

Abstract **- Word search puzzles significantly enhance vocabulary, spelling, visual scanning, concentration, and problem-solving skills. They provide an enjoyable method for learners of all ages to practice new words or concepts, particularly in language acquisition. The study explores the design of a word search game, where words are dynamically selected from a dictionary and arranged into a grid. It highlights cognitive benefits, user engagement, and educational potential. The results show that such games improve vocabulary and pattern recognition. It also discusses technical aspects, challenges in word placement, and user experiences in both digital and paper-based formats, offering insights for educational use in language learning.**

Keywords - Word Search Puzzles, Vocabulary Enhancement, Spelling Improvement, Visual Scanning Abilities, Concentration, Interactive Learning, Language Acquisition, Cognitive Benefits, Educational Potential, Digital Formats, Dynamic Word Selection, User Engagement, Language Learning Support

I. INTRODUCTION

Word search puzzles have long been a popular activity for enhancing cognitive skills such as pattern recognition, focus, and problem-solving. While traditional word search games have remained largely unchanged, there is a growing interest in introducing dynamic and interactive elements that enhance user experience. The motivation for developing this dynamic word search game stems from the desire to offer a more engaging and challenging experience, tailored to different skill levels. By dynamically generating the grid based on selected difficulty levels—easy (10x10), medium (13x13), and hard (15x15)—players can enjoy a fresh puzzle each time they play.

This game utilizes a hashtable-based dictionary for efficient word placement, ensuring that words are positioned both horizontally and vertically within the grid. Random characters fill the empty spaces to add complexity. A timer is also integrated, with varying time limits according to difficulty levels, providing an added challenge and fostering time-management skills. The final result is displayed with the player's answers highlighted in different colours: correct answers in green and incorrect ones in red. The game not only provides an entertaining experience but also promotes cognitive development by encouraging players to recognize words under time pressure. This paper presents the design and implementation of this dynamic word search game, aiming to showcase how interactive elements can enhance traditional puzzle games, making them more engaging and challenging for a wider audience.

Advantages of this Word search puzzle

1. **Customization of Difficulty Levels**: -The game offers three levels of difficulty (Easy, Medium, Hard) based on grid size.
2. **Dynamic Grid Generation**: -Each time the game is played, a new grid is generated by selecting words randomly from a dictionary.
3. **Increased Challenge Through Word Variations**: - Words are randomly selected from a 300 word dictionary.
4. **Clear Word Placement**: -The game only places the words horizontally and vertically this enhances focus of the player while solving as compared to a boggle.
5. **Coordinates for Word Location and displaying final grid**: - The inclusion of coordinates for both horizontal and vertical words helps players understand where each word begins and ends.

6. **Immediate Feedback**: - The game provides instant visual feedback by displaying correct words in green and incorrect ones in red.
7. **Score Tracking**: -Displaying the score at the end of the game helps players track their progress.
8. **Flexible Word Count**: -The number of words included in each difficulty level is well-balanced (5 for Easy, 7 for Medium, 10 for Hard).
9. **Large Word Dictionary**: -With a dictionary of about 300 words, the game offers a wide variety of possible word combinations.
10. **Simple Interface**: - The game's focus on displaying only the words in the grid with clearly marked coordinates ensures that it remains user-friendly, making it accessible even to younger audiences or beginners.

II. LITERATURE REVIEW

The paper [1] explores how word search puzzles enhance students' vocabulary, showing significant improvements in vocabulary scores compared to traditional methods. Puzzles also improve spelling, memorization, motivation, and engagement.

Paper [2] discusses a digital educational game, Wordsearch, designed for learning foreign language vocabulary. Testing with 11 students showed its effectiveness in vocabulary retention and fostering intrinsic motivation due to its user-friendly interface.

Paper [3] examines the use of word search and crossword puzzles in nursing education. With 96 students in the quantitative phase, the study found that puzzles improved learning, motivation, and information retention, with students noting benefits such as increased engagement and topic comprehension.

Paper [4] reviews the impact of gamification on student motivation and engagement in education, analyzing 46 studies from 2016 to 2019. It highlights contradictions in the literature and the need to improve theoretical frameworks and implementation methods.

Paper [5] details the creation of a Word Search Puzzle using a multi-linked list data structure. The study found that this method offers superior efficiency for grid creation and data handling compared to existing models.

III. METHODOLOGY

To make the game play more interactive and to stand out against the traditional word search games we have incorporated the following technologies:

1. Hashtable: Efficient word storage and retrieval from the dictionary.
2. 2D Array: Visual grid representation where words are placed horizontally and vertically.
3. ArrayList: Handles user inputs and compares them to the grid's words.
4. Timer: Tracks time based on the difficulty level (1 minute for easy, 3 minutes for medium, and 5 minutes for hard).
5. Hash Set: To find words present in the grid from the dictionary and store the answer from users after accepting them in the stack to avoid the same word.
6. Stack: Accepts the input from user, usage of stack provides the undo feature to the user in case he/she enters wrong spelling or word.

The word search game dynamically generates the grid based on the selected difficulty (easy, medium, or hard) with grid sizes of 10x10, 13x13, and 15x15, respectively. Empty spaces are filled with random characters, ensuring unique puzzle generation each time. Upon completion, the game displays correct answers in green, incorrect ones in red, and replaces random characters with dashes ("-"). This combination of technologies creates an engaging and interactive gameplay experience.

Classes in the game:

1. Dictionary class: This class comprises the various words that will be used to generate the grid of words for various levels. The words are stored with the help of hashtable
2. Boggle Solver class: This class will generate a word grid depending upon the level selected by the user. This class will take words from the dictionary and place them in the grid by checking whether the word can be placed there or not. The word gets placed randomly, in horizontal or vertical ways. The remaining spaces are filled with random alphabets. After generating the grid it is displayed to the user. The timer gets started once the grid is displayed. The grid is then solved by the compiler. Scores are calculated based on answers accepted from users compared with the answers generated present in dictionary. The final grid with only words is displayed, wrong answers are displayed in red colour while correct ones are displayed in green colour.

A. Approach

To study the design and effects of a dictionary-based word search game, that is created as a console-based game prototype, dynamically generates a grid of letters by selecting words from a large dictionary and placing them randomly in the grid. The user is tasked with finding the words within a set time limit, depending on the difficulty level chosen.

B. Game Design

The game consists of two main components: the Dictionary and the Boggle Solver.

Dictionary Implementation: The Dictionary uses a hash table with 40 buckets to store words. A hash function calculates the index for each word according to the sum of the ASCII values of its characters modulo 40. The 'insert' function adds words to the hash table, while 'print_hash_table' displays the contents of each bucket.

Boggle Solver: The Boggle Solver class is responsible for generating the game grid and solving the word search puzzle. It includes several key functions:

1. **CreateArrayList**: This function initializes the dictionary and prepares a list of words based on the selected difficulty level.
2. **CreateBoggleGrid**: This function creates a game grid and places words in it randomly, filling the remaining cells with random letters.
3. **PlaceWordInGrid** and **canPlaceWord**: These functions handle the placement of words in the grid, ensuring they fit within the grid and do not overlap improperly.
4. **FillRandomLetters**: This function fills empty grid cells with random letters.
5. **SearchWord**, **checkHorizontal**, and **checkVertical**: These functions search for words in the grid, verifying their presence horizontally or vertically.
6. **Useranswers**: This function handles user input, allowing players to enter words they find in the grid and providing

IV. IMPLEMENTATION

Traditional word search games rely on fixed matrices, limiting their uniqueness and engagement potential.To enhance replayability and cognitive stimulation, a word search game is developed with a dynamic algorithm that randomly places words, creating varied and innovative gameplay experiences. The figure below indicates the grid of the system which is implemented in three levels: Easy, Medium and Hard.

Grids for Easy:

```
A X X W E S V N O L
C Y P X E N E R G Y
R O O G X Q T W A F
Y R E O T H G J O X
L G R M T W I L L N
I A V P G M R G X K
C N F J E P O C H G
Z T O M A X N U F E
T I L Y F C V A A G
B P F E Q J S Y N P
```

Medium:

```
A C X L V C Y T C Z U B Q
V T A Z V G G V P U W N O
T Q M A P S I M O Z V S U
K R D C L T H X B I P G I
I O S E A J U W Y P B E E
X B K Y N A B Q G P L H N
Y Q T W E B T K G E P Q M
R N B D T O B Y N R L B X
W F N T B T I A R G E I F
H S Y O P D R R L F X H O
V L Q B H W R E J N K B R
N M U I S Y T A J B U T C
D W A T T A C H C U R L E
```

Hard:

```
M N N D L X D U T U B M R L G
Y V G O C O N C E P T U A L R
T O I W W B Y M P W G N M Y C
S F K S I H S E Q U I N U T V
T L A K B M P M W R L O L B K
R A T V F S O V K D Z O J K I
Q N F Z O V E U K C G R C V C
T N M K R Z C Q S B C O E E S
R E A R C M X Y R E W V L L L
K L X Y E Q D K W D X Y L V N
M A C R A M E I B L F G Q E R
W A F C X P A I S L E Y E T T
M T H R E A D C P W T L R E O
L I Q A Z R T O A D K U W B P
S T M L K X U G U C A B Z W Q
```

Figure 1: Grid of three different levels

Game play for hard level:

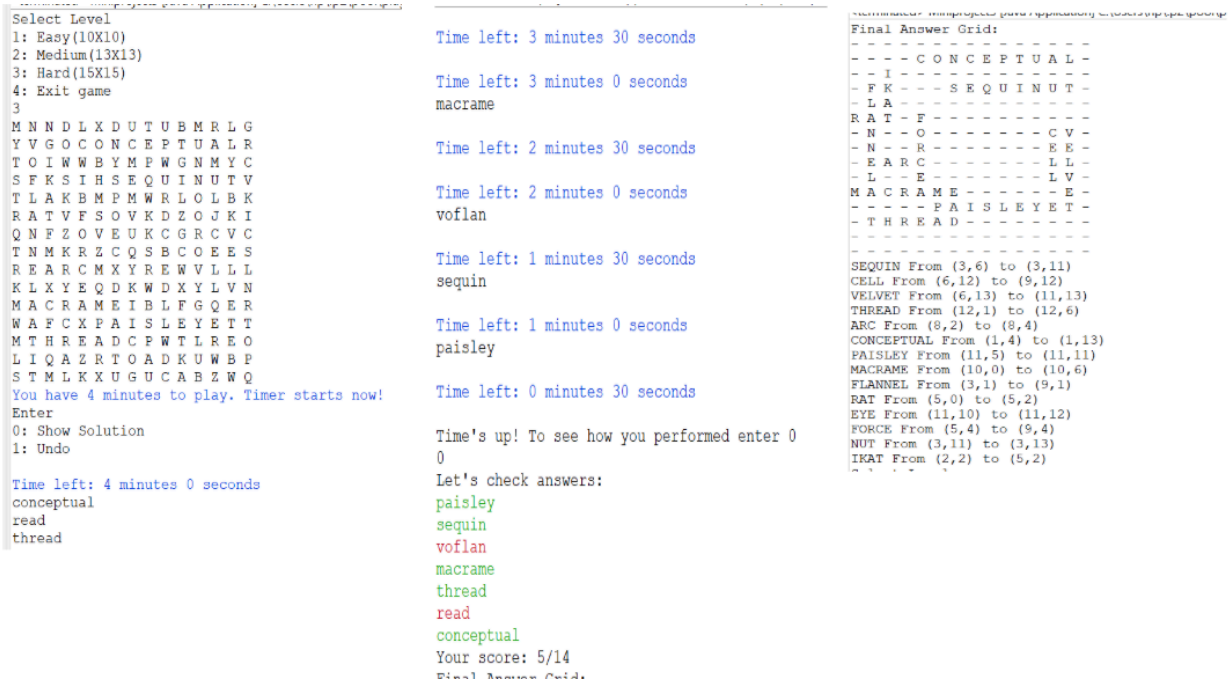

Figure 2: Levels of game

A. Game Performance: The overall effectiveness of the application is most impacted by the performance of search(). When utilizing a hashtable, this function has a time complexity of O(size^2 * d * l), in which "size" stands for the grid dimension, "d" is the dictionary size and "l" is the average word length. The grid generation runs at O(size^2). Processing the user's input is linear to the input count. The corresponding space complexity with the hashtable is O(N+M), where N is the hashtable size and M is the total number of words.

B. Impact of Database Integration: Performing the search function for the dictionary is greatly affected by the implementation of a dictionary storage database. Assuming the use of indexed columns, database queries would execute in O(logN) time, with N being the number of entries in the dictionary. This change would greatly improve overall search time, especially for bigger dictionaries. The space complexity regarding the application's reset footprint would decrease to O(1), now that the dictionary has to be moved to a disk with O(M*l) space usage, instead of RAM.

C. Cognitive Benefits: There is expected to be a significant rise in the cognitive levels of the people playing this game. It will improve their vocabulary and will develop a habit of pattern recognition. It will make their brains sharper as the time limit feature increases their ability to think fast. A good vocabulary will in turn improve their soft skills.

V. ANALYSIS OF RESULT

There are few questions which were asked to the parents and to the students who played this game search. These are a few questions which were asked to them by covering different aspects like Frequency of Game play, Pattern recognition, Vocabulary, Concentration, skills improvement, engagement with Game etc. Sample questions are mentioned below:

FreqPlay - How often do you play word search puzzles?

RecogPattern - How confident are you in spotting patterns or hidden words in a puzzle?

Vocab/Spell - Do you feel the word search helps improve your vocabulary or spellings?

Focus/Conc - How would you rate your focus and concentration while solving puzzles?

ImproveSkills - Do you feel the word search improved your problem-solving skills or cognitive abilities?

GameEngage - How engaging or immersive did you find the game experience overall?

Rate - How will you rate the technique of grid creation?

The bar chart given below in figure 3, presents a comparative study on the effectiveness of a method by measuring performance before and after its implementation. The x-axis represents different categories: FreePlay, RecogPattern, Vocab/Spell, Focus/Conc, ImproveSkills, GameEngage, and Rate, while the y-axis quantifies the effectiveness on a scale from 0 to 5. The chart uses two shades of pink to differentiate between "Before" and "After" conditions. The "After" bars are consistently higher across all categories, indicating an improvement in effectiveness after applying the method. The most significant improvements appear in RecogPattern, Focus/Conc, and ImproveSkills, suggesting the method had a strong positive impact in these areas.Overall, the study suggests that the method tested led to an increase in effectiveness across all evaluated aspects.

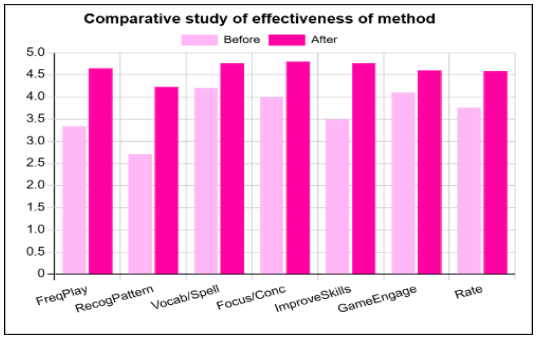

Figure 3: Comparative study of effectiveness of method.

The pie chart in figure 4, illustrates responses to the question "How likely are you to use word search for educational purposes?" The majority of participants (34.2%) responded "Somewhat likely," followed by "Very likely" (26.2%), indicating a generally positive attitude toward word search for education. A smaller proportion of respondents were neutral (19.3%) or unlikely/very unlikely (11.9% and 8.4%) to use it for educational purposes.

The pie chart in figure 5, illustrates responses to the question of whether dynamic word placement made a word search game more or less enjoyable compared to traditional word searches. The majority of respondents (68.5%) found the game more enjoyable, while 25.5% felt it was about the same, and only 8% considered it less enjoyable. The data suggests that dynamic word placement positively impacts user enjoyment.

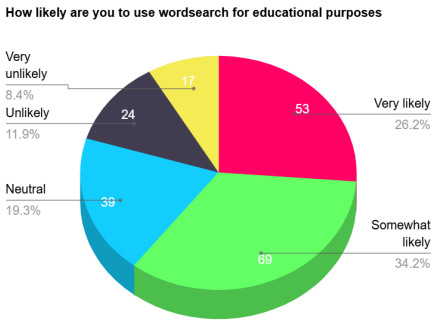

Figure 4: Use of Word search

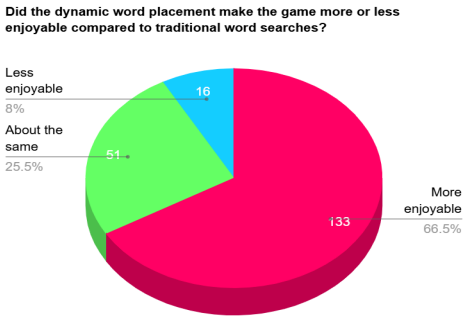

Figure 5: Game entertainment stats.

VI. TECHNICAL CHALLENGES

The primary technical challenge in designing the game was ensuring the efficient placement of words in the grid. Word placement algorithms must be optimized to avoid overlapping words while ensuring that all words are placed within the grid without excessive empty spaces. We added a minimum of 5 words in easy level, 7 words in medium level and 10 words in hard level. The count of words may increase because of the words which are generated due to placement of random alphabets

in the remaining spaces. Another technical challenge was to integrate timer in the game as each level had different times for solving and some players may quit the game in between by entering 0. Future improvements could involve more sophisticated algorithms for word distribution and difficulty scaling.

VII. CONCLUSION

The word search games, when designed with a dictionary-generated word grid, can have significant cognitive and educational benefits. Such games provide a dynamic and engaging way to improve vocabulary, memory, and pattern recognition skills. The findings suggest that this type of game could be particularly useful in educational settings, such as schools or language learning apps. Further research could explore additional features such as multiplayer modes or integrating different languages to broaden the game's educational scope.

VIII. FUTURE WORK

The dictionary present now is implemented using hashtable, we want to implement it using a database and adding more words to it. This game can be used to fill the gap between learning and playing as you can learn new words while playing this game. It can be useful for children from age 5 to 20 (that is standard UKG to graduation) as it can be used to learn new terms easily. There would be two options- learn new words (for the users wanting to play the game just for fun) and curriculum based (for students). We can integrate AI too. When the student enters his/her class and topic (school student), or a college going user enters his degree, subject and topic, or someone wanting to play just for fun enters the domain name, AI would select the words of desired degree and subject or, class and subject, or domain from the database and generate a grid of easy level at first. As the user ascends through the game, AI would recognize the pattern of user progressing and it would dynamically level up if the user is progressing well and if the user is finding it difficult, then it would keep generating easier level grids so that the the user gets comfortable with solving easy level grids first, later increasing the toughness.

IX. REFERENCES

[1] T. N. Fitria, "The effectiveness of word search puzzles game in improving student's vocabulary: A systematic literature review," *Pioneer: Journal of Language and Literature*, vol. 15, no. 1, pp. 50-67, 2023.
[2] Goumas, Stefanos; Terzopoulos, George; Tsompanidou, Despina; Iliopoulou, A. Word Search, an Educational Game in Language Learning Goumas, 2020.
[3] Patel, Shivam; Mahariba, A. Word Search Puzzle using Multi-Linked Lists, 2020
[4] Nevra Kalkan, Sevil Güler, Hülya Bulut, Ali Ay, "Views of students on the use of crossword and word search puzzle as a teaching technique in nursing education: A mixed-method study, Nurse Education Today", Volume 119, 2022,
[5] Zainuddin, Zamzami & Chu, Samuel & Shujahat, Muhammad & Perera, Corinne. (2020). The impact of gamification on learning and instruction: A systematic review of empirical evidence.
[6] Larsen, C., & Butler, M. (2009). Educational puzzles and games for enhancing vocabulary skills. Educational Technology Review, 16(3), 245-258.
[7] Miller, R., et al. (2014). Cognitive benefits of word games: A systematic review. Journal of Cognitive Development, 22(4), 101-110.
[8] Smith, J., & Jones, S. (2017). Game design and dynamic difficulty adjustment. Journal of Interactive Media, 25(2), 33-48.
[9] Sartaj Sahani, "Data Structures, Algorithms and Applications in JAVA", Universities Press (2 nd edition).
[10] Robert Lafore, "Data Structures Algorithms in JAVA", Techmedia,(1 st edition).
[11] E. Horowitz, S. Sahni, D. Mehta, "Fundamentals of Data Structures in C++", Galgotia Publications ,(2 nd edition).
[12] Yedidyah Langsam, Moshe J Augenstein, Aron M Tenenbaum, "Data Structures using C and C++" , Pearson Education, (2 nd edition).
[13] A. Aho, J. Hopcroft, J. Ulman, "Data Structures and Algorithms", Pearson Education, (2 nd edition).
[14] Brassard and Bratley, "Fundamentals of Algorithmics", Prentice Hall India/Pearson Education, (2 nd edition) .
[15] M. Weiss, "Data Structures and Algorithm Analysis in JAVA", Pearson Education (3rd edition), (2012).
[16] Goodrich, Tamassia, Goldwasser, "Data Structures and Algorithms in JAVA", Wiley publication, (6th edition)
[17] R. Gillberg, B. Forouzn, "Data Structures: A Pseudocode approach with C", Cengage Learning, (2 nd edition).
[18] Backlund T, Hendrix M, "Educational games- are they worth the effort? A literature survey of the effectiveness of seroius games".
[19] Dr R. Satheeshkumar, Sachana C, Dr S Gokula Krishnan, Gururaj FD, "Role of gamification in the Indian education system with reference to higher education institutions in Bangalore city".
[20] Darina Dicheve, Christo Dichev, Gennady Agre, Galia Angelova, "Gamification in Education: A systematic mapping study".