Model

Controller

View

**NodeList**

- next: Node
- data: T

+ getNext(): Node
+ setNext(next: Node)
+ getData(): T
+ setData(data: T)

**LinkedList**

- size: int
- first: Node<T>
- last: Node<T>

+ insert(data: T)
+ insertAtFirst(data: T)
+ set(index: int, data: T)
+ get(index: int): Node<T>
+ delete(index: int)
+ contains(data: T): boolean
+ toString(): String
+ isEmpty(): boolean
+ getSize(): int
+ getFirst(): T
+ getLast(): T

**Vertex**

- data: T
- edges: LinkedList<Vertex>

+ toString(): String
+ getData(): T
+ setData(data: T)
+ getEdges(): LinkedList<Vertex>

**Graph**

- maxVertices: int
- size: int
- vertices: Vertex<T>[]
- currentVertices: int

+ addVertex(T data): int
+ addVertex(data: T, index: int)
+ addEdges(from: int, to: int)
+ replaceVertexData(index: int, data: T)
+ degree(index: int): int
+ toString(): String
+ isEmpty(): boolean
- verifyIndex(index: int)
- searchNullIndex(): int
+ getMaxVertices(): int
+ getSize(): int
+ getCurrentVertices: int
+ getVertices(): Vertex<T>[]

**WordSearchModel**

- callFromRoot(int algorithm, Graph graph, char[] word): boolean
+ dfsSearch(Graph graph, char[] word): boolean
- dfs(char[] word, int letter, Vertex root, boolean[] visited, boolean search): boolean
+ bfsSearch(Graph graph, char[] word): boolean
- bfs(Vertex root, char[] word, boolean[] visited): boolean

**WordSearchGraphDisplay**

- nodeCount: int

+ getNodeCount(): int
- callFromRootAndDisplay(int algorithm, Graph graph, char[] word): boolean
+ viewDfs(Graph graph, char[] word): boolean
- dfsLib(SingleGraph graphView, char[] word, int letter, Vertex root, boolean[] visited, boolean search): boolean
+ viewBfs(Graph graph, char[] word): boolean
- bfsLib(SingleGraph graphView, Vertex root, char[] word, boolean[] visited): boolean
- addEdge(SingleGraph graphView, Vertex src, Vertex dst)

**Queue**

- queue: LinkedList<T>

+ getHead(): T
+ enqueue(T data)
+ dequeue(): T
+ isEmpty(): boolean
+ toString(): String
+ getSize(): int

**App**

- soupGraph: Graph<Character>
- dictionary: LinkedList<String>
-msTime: double

+ main(String[] args)
+ sendFilePath(String path): boolean
+ searchWord(int choice, String word): int
+ searchWords(int choice): boolean []
+ addDictionaryWord(String word): boolean
+ updateDictionaryFile(): boolean
+ getDictionary(): LinkedList<String>
+ getGraph(): Graph<Character>
+ getMsTime(): double
+ wordsToChars(): char[][]

**FileController**

- filePath: String

+ read(String path): String[][]
+ write(LinkedList<String> dictionary, Graph graph): boolean
+ isValidWord(String word): boolean

**GraphController**

+ createGraph(Character[] letters): Graph<Character>
- createEdges(Graph graph)

**FileSelector**

**- chosenFile: File**
**- result: int**

**+ continueButtonActionPerformed()**

**Menu**

- dictionaryModel: DefaultListModel<String>
- dictionaryCellsColor: LinkedList<Integer>

**Help**

- menuGUI: Menu

+ Help(Menu menuGUI)