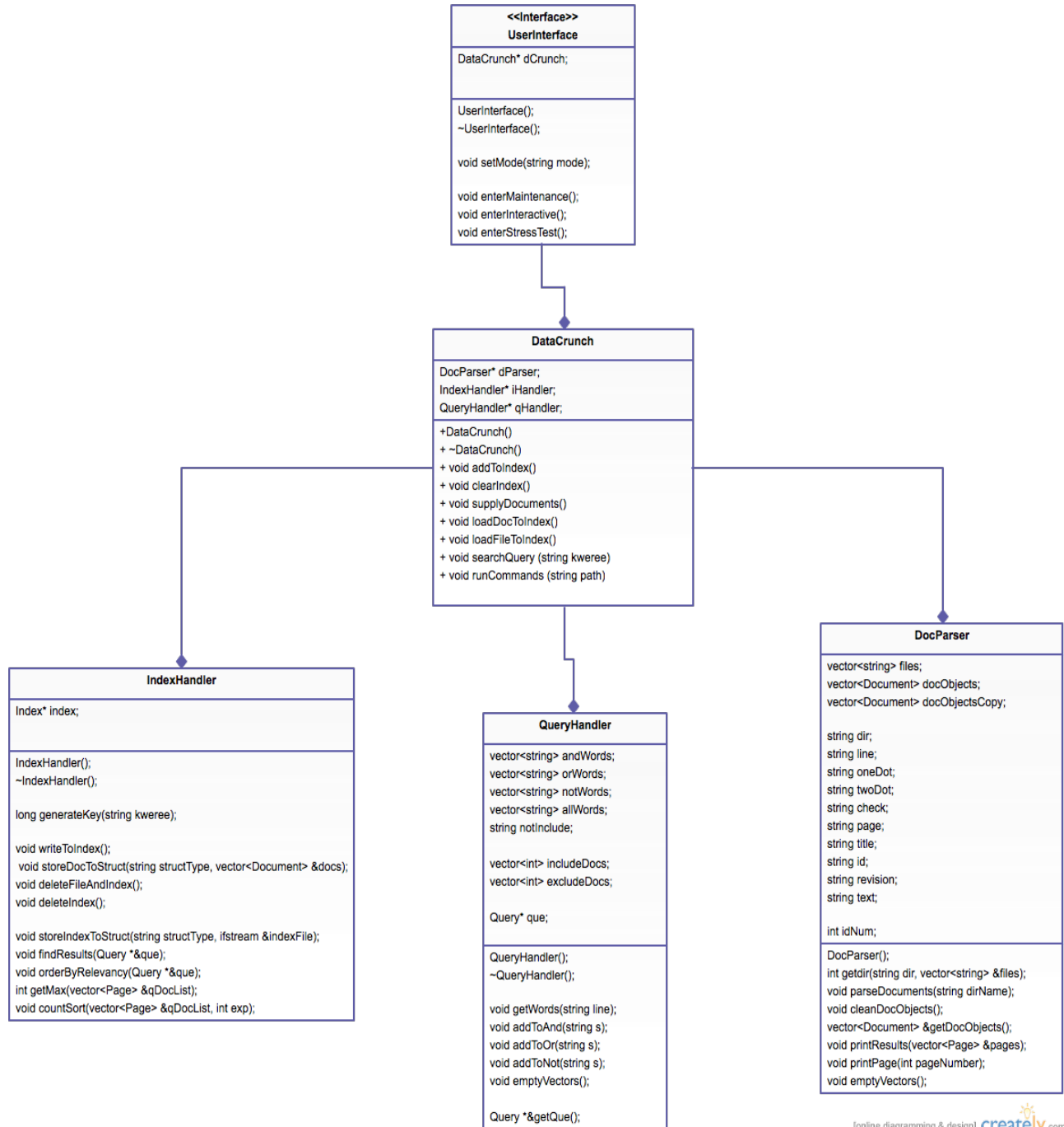


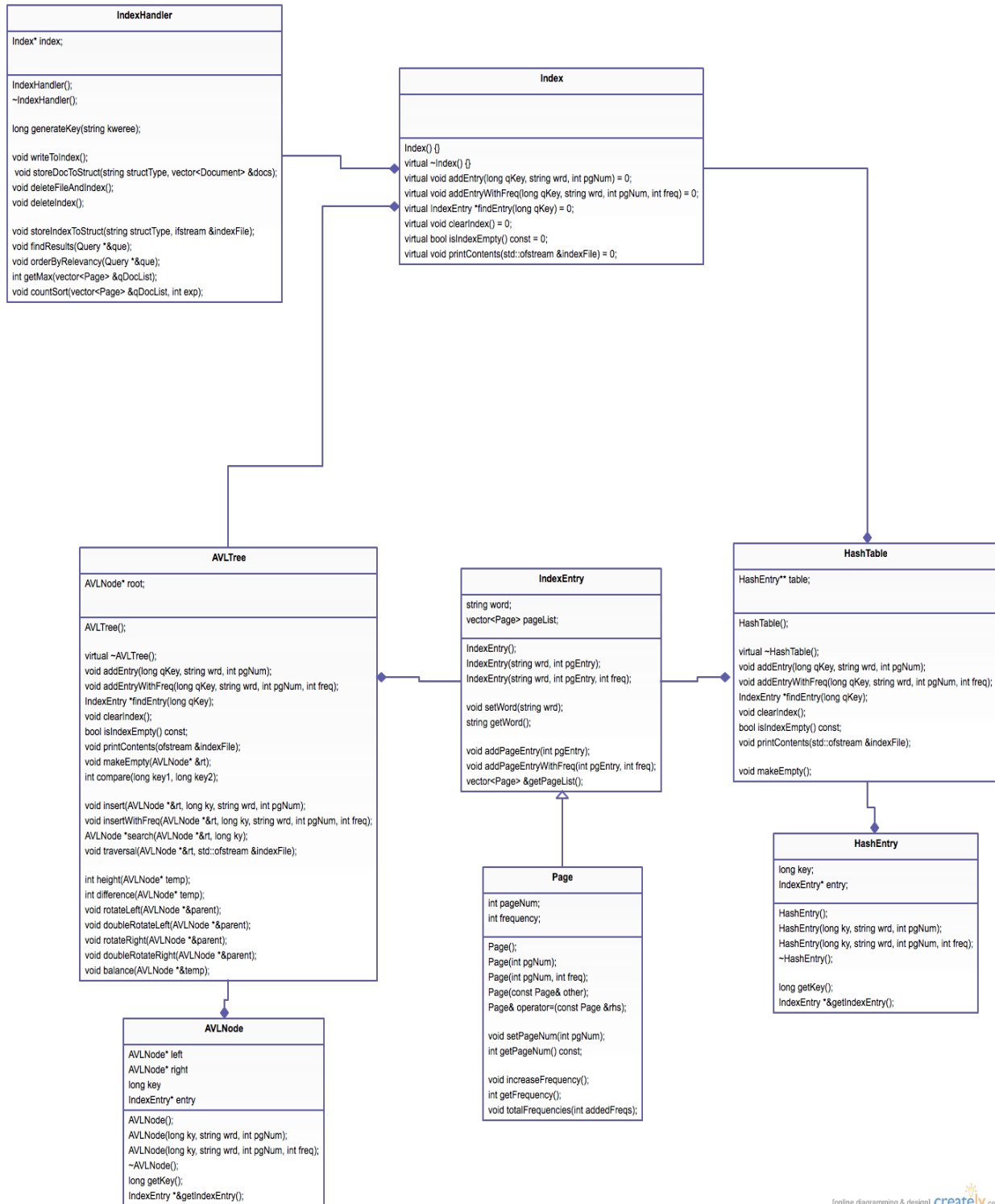
Team Zombie Coders

Members: Aurora Havens and Emely Villeda-Principe



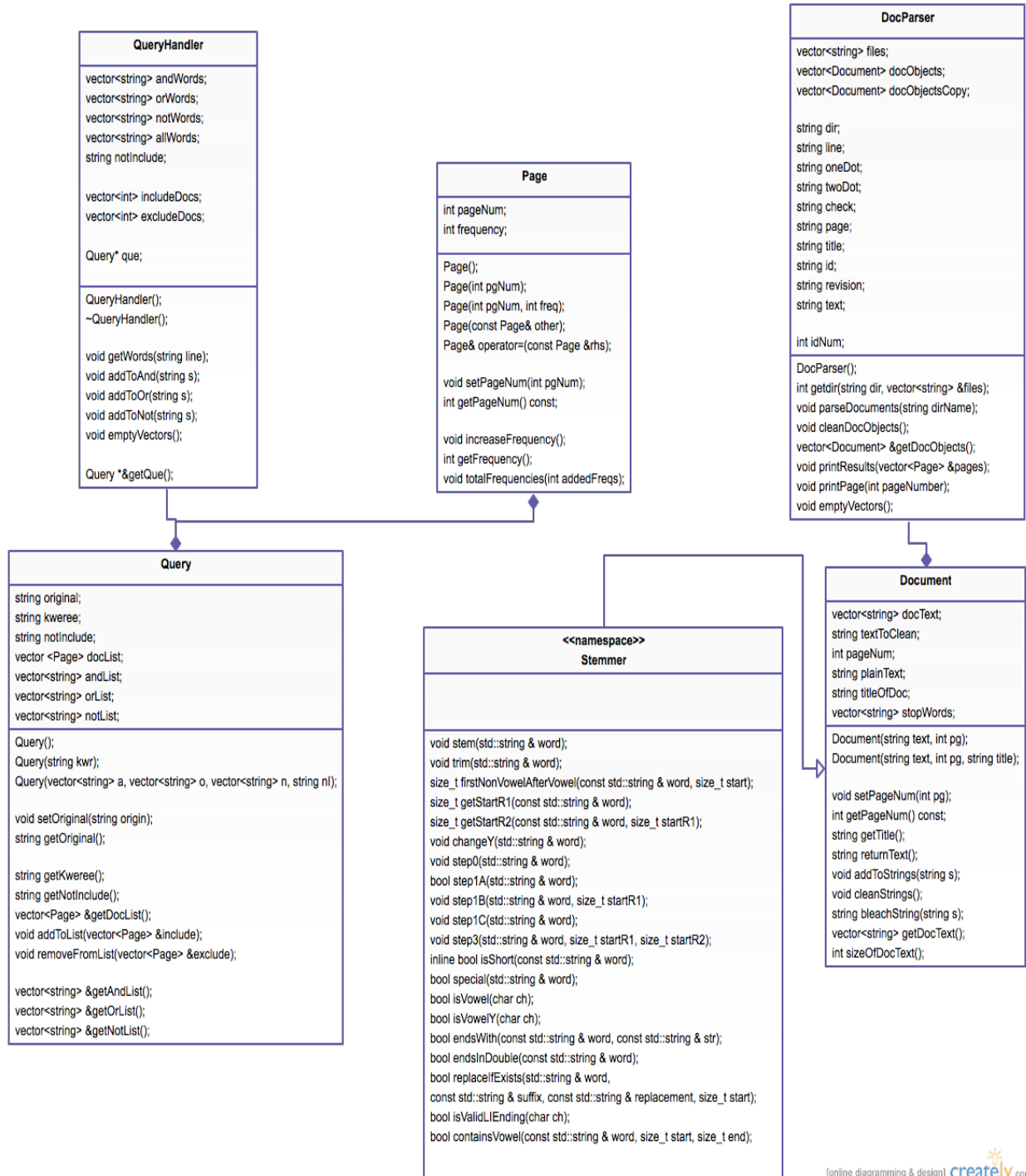
Team Zombie Coders

Members: Aurora Havens and Emely Villeda-Principe



Team Zombie Coders

Members: Aurora Havens and Emely Villeda-Principe



Team Zombie Coders
Members: Aurora Havens and Emely Villeda-Principe

Class Roles and Responsibilities

- **Class UserInterface [Owner: Emely]**
 - Acts as the connector from main to DataCrunch in order to start the program
- **Class DataCrunch [Owner: Aurora]**
 - Accesses the 3 main processes that run the search engine (DocParser, IndexHandler, and QueryHandler)
 - Handles functions required for completing user requests
- **Class DocParser [Owner: Aurora (Including Document)]**
 - Takes in XML Documents to parse data to store relevant information
 - Responsible for retrieving specific data from the file and creating document objects with strings and pages numbers in side of them
- **Class Document**
 - Creates an object that will be stored out into the index so we can access the string and the documents that string is located in
- **Class IndexHandler [Owner: Emely (Including Index and its implementation classes)]**
 - Handles writing an index onto disk as well as storing the index data into an accessible data structure
 - Responsible for retrieving entries from the index and returning them to the user
- **Class Index**
 - Serves as the interface of the data structure that will hold the index data
 - Composed of index entries and the structure is dependent on the user
- **Class IndexEntry**
 - Serves as an entry containing the word and list of documents that include the term
 - For storing in a data structure that would allow for easy searching upon user request
- **Class AVLTree**
 - Serves as a potential data structure for storing index entries
 - Composed of nodes that are accessible to the user
- **Class AVLNode**
 - Node for the AVL Tree data structure
 - Holds IndexEntry object containing a string and a vector of document numbers
- **Class HashTable**
 - Serves as a potential data structure for storing index entries
 - Composed of HashElements that are accessible to the user
- **Class HashEntry**
 - Element for the Hash Table data structure
 - Holds IndexEntry object containing a string and a vector of document numbers
- **Class QueryHandler [Owner: Aurora (Including Query)]**
 - Parses through query requests to determine what the IndexHandler must find
 - Responsible for interpreting the parts of a query (terms and Boolean operators)
- **Class Query**
 - Stores the parts of a query (terms associated with each Boolean operator)

Team Zombie Coders

Members: Aurora Havens and Emely Villeda-Principe

- Necessary for storing relevant documents once the IndexHandler finds matches
- Class Stemmer **[Owner: Aurora]**
 - Responsible for stemming the words and getting rid of stop words
- Class Page **[Owner: Emely]**
 - Responsible for holding a page's page number and it's frequency

Team Zombie Coders
Members: Aurora Havens and Emely Villeda-Principe

Set of initial commands that are used in stress test mode to test the functionality of the program:

1. mmode
2. add
3. new
4. default
5. avl
6. return
7. imode
8. load
9. avl
10. search
11. main
12. return
13. mmode
14. clear
15. add
16. new
17. default
18. hash
19. return
20. imode
- !