CRC cards

WebScraping	
Instance variables: referrer (http://google.com) userAgent (Mozilla)	Collaborators: NLP Analysis (feed input)
Methods: sanitizeURL	Make sure URL is in proper form for feeding into JSOUP parser
getConnection	Establish connection to website, throw exceptions as necessary if unable; return html if connection
cleanContent	Get only relevant html tags, clean content so that content cannot be malicious to application
readHTML	Read in relevant tags to string array for output
Runner	Solicit user input, run methods on input url, return for passing into NLP

Model Training	
Instance variables: *still learning how this works; will train for keyword and topic tagging on a dataset 1. Acquire data 2. Clean data 3. Train the model 4. Evaluate the model	Collaborators: Dataset NLP Analysis

NLPAnalysis	
Instance variables: userWords (string of words from user’s URL) InputStream tokenModelIn InputStream posModelIn InputStream dictLemmatizer (?) tokensArray (string array of tokens) tagsArray (string array of tags) HashMap<String, String> tokenToPOSTagMap HashMap<String, Integer> tokenToCountMap	Collaborators: WebScraping (output) File: token model (pre-trained) File: pos model (pre-trained) File: a lemma dictionary (from OpenNLP) -likely needed for KeywordAnalysis Sentiment Analysis (input) KeywordAnalysis
Methods: Constructor - takes in string from web scraping output (userWords) Lematize (helper) Return tokensArray createTokenToPOSTagMap Return tokenToPOSTagMap createTokenToCountMap Return TokenToCountMap nlpLemmatize Return lemmaArray	Responsibilities: Creates the NLPAnalysis object Lematizes the string input from the user Tokenizes, POS tags, stores those key-value pairs in hashmap Put all words (minus function words) and their frequency of occurrence (from user’s URL) into a hashmap for the sentiment analysis Create string array of lemma of each word

getTokenToCountMap	Getter for Sentiment Analysis and Keyword Analysis to access

Keyword Analysis	
Instance variables: keywordArray InputStream keywordModelIn	Collaborators: NLPAnalysis (input) User Interface (output) File: text keyword model (from our Model Training)
Methods: Constructor takes a String of the words from the user's URL (userWords) nlpKeywordTag Return keywordArray	Responsibilities: WebScraping (input) Use our trained model to find keywords and thus identify the topic(s) of the text

Sentiment Analysis	
Instance variables: negativeWordCount positiveWordCount scoreOutput	Collaborators: NLP Analysis (input)
Methods: createDictionary wordCounter	Responsibilities: Create a dictionary of positive and negative words Count the positive and negative words in the text

scoreCounter scoreDisplay → go to user interface??	Label the positive words with grade 1 and the negative words with grade -1 Output a positivity score of the text

User Interface	
Instance variables:	Collaborators: NLP Analysis & Sentiment Analysis
Methods: Simple GUI? Print a file? Console?	Responsibilities: Returns to user the topic (several keywords) of the text from their URL input Optional (if time): recommend other URLs based on that