CSE611 Online Dictionary Technical Design Document

1. Tech Stack

Front End:

· ReactJS: React is a free and open-source front-end JavaScript library for building user interfaces based on components. It will be helpful to use react in our project, for implementing an interactive UI. It also is quick to implement and includes reusable components which will be helpful during development.

Back End:

· NodeJS: It works well with react and also has good potential for scalability. Node JS will be used to bind all the databases which we will be using.

· Express: It is a framework that is used with NodeJS. It is lightweight and one of the most popular frameworks.

Languages:

· JavaScript

· Python

Database:

· MySQL : Most popular in terms of security and dependability. It is fast ,highly scalable and efficient.

Source-Code Control:

· GitHub

Libraries, Tools, APIs:

· NLTK: Natural Language Toolkit, a library in python for NLP

· Google TTS: Speech services API

· Dictionary API

· VSCode

- Stackify (Logging)

System Requirements:

- Basic

2. Accounts and Infrastructure

2.1 Development and Test

- AWS

AWS EC2 Free Tier Instance is being used for this project.

2.2 Production

- IBM

IBM Servers provided by Project Faculty.

Data Sources, Models, Timing

1.1 Data Sources

- **WordNet** - WordNet is a large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations
This license is available as the file LICENSE in any downloaded version of WordNet.
Wordnet is free to use and royalty free.

- **Webster (Licensed -Gutenberg) –**
https://github.com/matthewreagan/WebstersEnglishDictionary
The JSON files can be used as-is. Any project which can parse JSON should be able to read and process these files

License - The original dictionary text file is covered by The Gutenberg Project's licensing, please see the file headers for more details. The Swift parsing tool and example output files in this repository are free and distributed under the GNU General Public License, Version 2.

- **NY Times and other news sources (Scraping data)**

The Data can be scraped through NY Times by using newsanchor package and AFINN dictionary. The scraped data can help gather sentences based on the words hit by the user. Yellowpages.com can also be used as a resource to gather data.

License - Developer Network of NY Times has information regarding licenses. We can use data from the NY Times as long as the data is cited correctly.

- **Dictionary API** - https://www.dictionaryapi.dev/

License - https://github.com/meetDeveloper/freeDictionaryAPI/blob/master/LICENSE

1.2 Data Models and Structure

The online dictionary will be primarily supported by data from Wordnet, and Webster dictionary. Wordnet can be used with the help of inbuilt python libraries like NLTK. Data from Webster, after reading from the database, will be converted to JSON format for further processing.

There are going to be two databases in the SQL server. The Main database is for storing dictionary data and the Logging database is for storing logs.

1.2.1 Main Database

Webster Dictionary table:

Table consists of following columns

Word - string denoting the dictionary term. Column does not allow null values.

PartsofSpeech - string denoting the parts of speech of the word. Null values are allowed.

Definition - The definition of the word in the specific parts of speech. Column allows Null values.

API Response table:

This table's primary purpose is to log the API's responses as the dictionary is continually used. The API will not be queried for a word that exists in this table.

Word - string denoting the dictionary term. Column does not allow null values.

PartsofSpeech - Parts of speech of the word returned by the API. Column allows Null values.

Definition - Definition of the word returned by the API. Column allows Null values.

ExampleUsage - Usage of the word in a sentence returned by the API. Column allows Null values.

SearchTime - Datetime value to check whether captured API response is up to date.

Pronunciation table:

This table is used to cache audio pronunciation of words. Before querying the API, we can query this table first to check if pronunciation audio for a word exists. Time and bandwidth is saved.

Word - string denoting the dictionary term. Column does not allow null values.

PartsofSpeech - Parts of speech of the word returned by the API. Null values are not allowed.

AudioLink - Link for the audio file of the word pronunciation returned by the API. Null values are not allowed.

1.2.2 Logging Database

Log table:

Word - string denoting the dictionary term. Column does not allow null values.

PartsofSpeech - Parts of speech of the word returned by the API. Null values are not allowed

WordFound - boolean indicating whether or not the word was found in any of the dictionary datasources. Column does not allow null values

SearchTime - datetime showing the time the search was queried. Column does not allow null values

IsAPIResponse - Boolean value showing whether a user query was responded to with the API or with the local data sources.

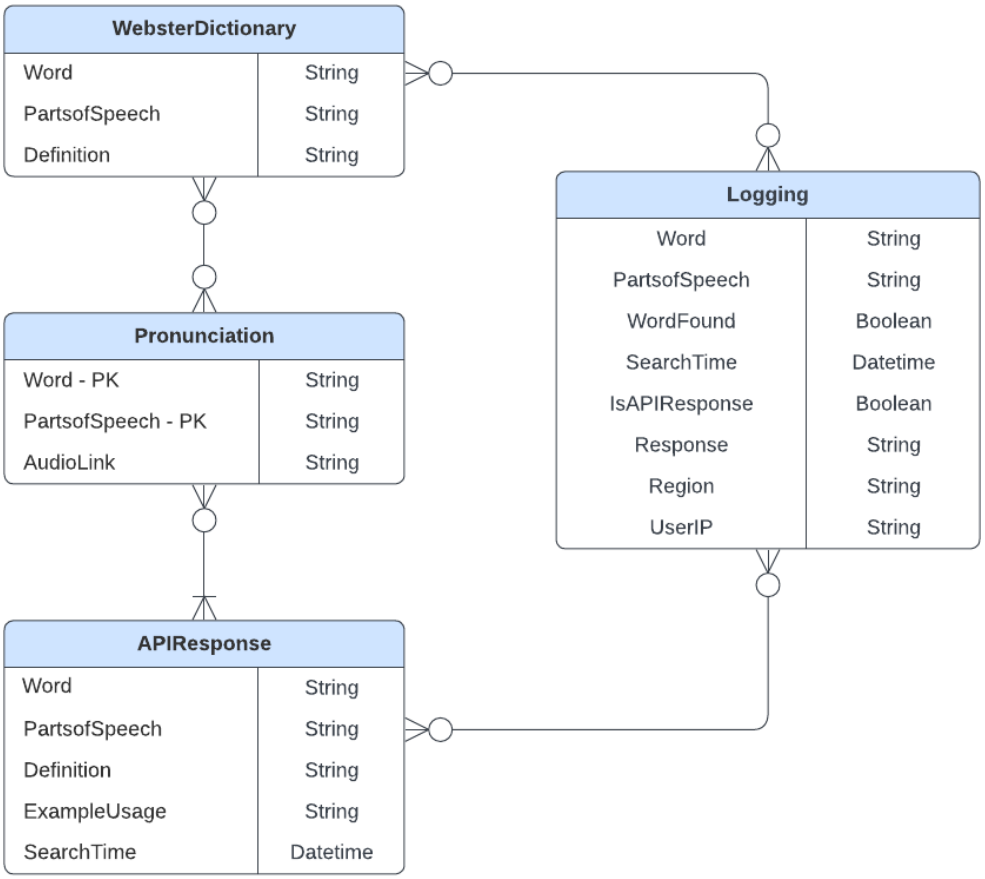
Response - string. Captures the response shown to the user. Useful for debugging. Column does not allow null values.

Region - string - Region of the world where the user is located. Column allows Null values.

UserIP - string - IP address of user. Useful for tracking and security. Column allows Null values.

1.2.3 Entity Relationship Diagram:

ER Diagram for Online Dictionary application.

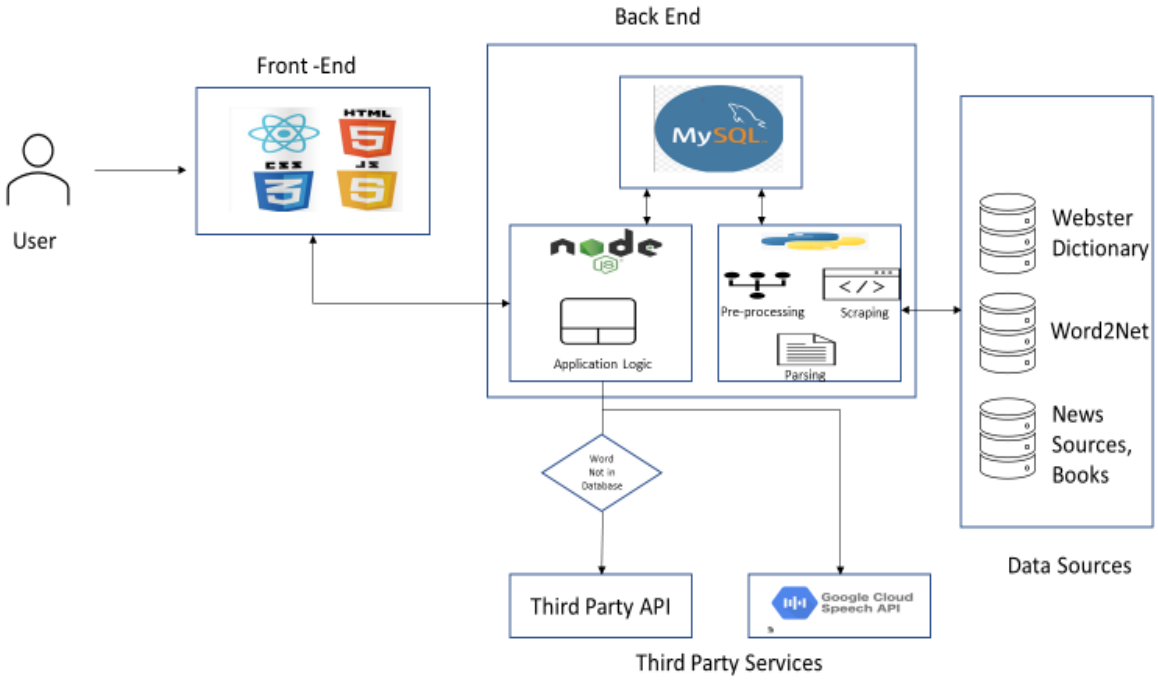


1.3 Timing

Whenever the user searches a word it is stored in the log table in the database. The frequency of each searched word is also stored in the same table which can later be used for analytical purposes. The data is stored forever. If the database is deleted the application still works as we will be using the API calls to fetch the meanings and sentences when the database doesn't contain the required information.

The word for which meaning is not found is also stored separately and later if it is verified as a correct word the word,meaning and sentence is added to the database.This is done weekly basis.

System Architecture Diagram



Deployment Methodology

All the data from the development server, like the Database tables, audio files (if any), application code, packages, modules, will be moved to production servers. Production server will be on IBM. Resource details provided by the Faculty.