



Computing & Information Sciences

*Florida International University
School of Computing and Information Sciences*

Software Engineering Focus

Final Deliverable

Project: Vocabulary In Reading (VIR)

Team Members: Camilo Rivera and Charles Benitez

Product Owner(s): Seyedjafar Ehsanzadehsorati

Mentor(s): Eric Dwyer and Mohsen Taheri

Instructor: Masoud Sadjadi

The MIT License (MIT)
Copyright (c) 2016 *Florida International University*

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

Abstract

The information in this document is to explain VIR, the purpose, the design choices and the development. VIR is a web app which facilitates learning of new languages through analysis of text and gathering of data which is then displayed to the user. This data is important to the user as it categorizes words in the text and gives priority to more popular words. The user then knows which words to use more and makes it easier to use these in conversation. Data given to the user includes category of word as far as popularity and definition of the words in the text with respect to the information in the database.

Table of Contents

INTRODUCTION

.....	5
CURRENT SYSTEM	
.....	5
PURPOSE OF NEW SYSTEM	
.....	5

USER STORIES

IMPLEMENTED USER STORIES	
.....	7
PENDING USER STORIES	
.....	10

PROJECT PLAN

HARDWARE AND SOFTWARE RESOURCES	
12	
SPRINTS PLAN	
.....	13
<i>Sprint 1</i>	
.....	
..... 13	
<i>Sprint 2</i>	
.....	
..... 13	
<i>Sprint 3</i>	
.....	
..... 14	
<i>Sprint 4</i>	
.....	
..... 15	
<i>Sprint 5</i>	
.....	
..... 16	
<i>Sprint 6</i>	
.....	
..... 17	

<i>Sprint 7</i>	
.....	
..... 18	
SYSTEM DESIGN	
ARCHITECTURAL PATTERNS	
.....	20
SYSTEM AND SUBSYSTEM DECOMPOSITION	
21	
DEPLOYMENT DIAGRAM	
.....	22
DESIGN PATTERNS	
.....	22
SYSTEM VALIDATION	
.....	23
GLOSSARY	
.....	
...	37
APPENDIX	
.....	
...	38
APPENDIX A - UML DIAGRAMS	
.....	38
<i>Static UML Diagrams</i>	
.....	38
<i>Dynamic UML Diagrams</i>	
.....	40
APPENDIX B - USER INTERFACE DESIGN	
52	
APPENDIX C - SPRINT REVIEW REPORTS	
69	
APPENDIX D - USER MANUALS, INSTALLATION/MAINTENANCE DOCUMENT, SHORTCOMINGS/WISHLIST DOCUMENT AND OTHER DOCUMENTS	
.....	74
REFERENCES	
.....	
..	80

INTRODUCTION

Vocabulary in Reading (VIR) is a website application developed in order to facilitate the learning of a language by displaying data relevant to the user such as word categorizing and definitions.

Current System

At the moment, there is no current system in place that is like this web app. This is a new system for aid in learning new languages.

Purpose of New System

The purpose of the new system is to create the application that Project Owner Seyedjafar Ehsanzadehsorati designed. Programmers Charles Benitez and Camilo Rivera implemented the system using User Stories provided by Ehsan and created a database using provided Excel files.

USER STORIES

The following section provides the detailed user stories that were implemented in this iteration of the VIR project. These user stories served as the basis for the implementation of the project's features. This section also shows the user stories that are to be considered for future development.

Implemented User Stories:

User Story ID	#123
User Story Name	Landing Page GUI
Description	As a User I would like to choose which word list to see so that I can distinguish the word types.
Acceptance Criteria	<ol style="list-style-type: none">1. Designing with diagrams2. Learn and utilize tools and tutorials.3. Titles and text boxes set up4. Links set up5. Programming front end (Html and Angular)
Owner	Camilo Rivera
Sprint Assigned	1,2

User Story ID	#117
User Story Name	Set up MySQL database
Description	As a product owner I want to reliably store information for the VIR app on a database.

	Upload provided wordlists for VIRS app to SQL Server
Acceptance Criteria	Acceptance Criteria: <ol style="list-style-type: none"> 1. SQL Server configured 2. User data imported 3. Foreign key relationships implemented 4. Create administrative panel to view data (Added 2/7) 5. Update administrative panel to import / edit data (Added 2/7)
Owner	Charles
Sprint Assigned	1

User Story ID	#129
User Story Name	VIRS Admin Panel
Description	<p>As a product owner I want to manage the backend of my app, so that I am able to view and update the data independently of the implementation of the GUI of the app.</p> <p>Write administrative site to manage backend of app using PHP/MySQL. Allow for searching, filtering, sorting, updating, and deleting word families and their members.</p>
Acceptance Criteria	<ol style="list-style-type: none"> 1. Categorizer (Enhanced Text) 2. Category Viewer 3. Frequency page 4. Dictionary page

	5. Import/Export pages
Owner	Charles
Sprint Assigned	2,3,4,5

User Story ID	#124
User Story Name	K1, K2, Off-List Words pages
Description	As a user, I want to access the list of type K1, K2, Offlist words, so that I may acquire the definition and context.
Acceptance Criteria	<ol style="list-style-type: none"> 1. Outputs the word 2. Allows for the word to contain links 3. Shows the recorded frequency of the word. 4. Page can acquire data from the database. 5. Back end is routed to K1, K2, Off List
Owner	Camilo Rivera
Sprint Assigned	

User Story ID	#126
---------------	------

User Story Name	AWL Page
Description	As a user, I want to access the list of type AWL words, so that I may know its presence in the database.
Acceptance Criteria	<ol style="list-style-type: none"> 1. Outputs the word 2. Allows for the word to contain link to info. 3. Shows if the word is currently present in the database. 4. Page can acquire data from the database. 5. Back end is routed to AWL data
Owner	Camilo Rivera
Sprint Assigned	4

User Story ID	#128
User Story Name	Interface MySQL to Express/Node
Description	As an admin, I would like my app to know the categories of the words so that I can distinguish them and apply different actions on these words.
Acceptance Criteria	<ol style="list-style-type: none"> 1. Create queries to go through each word in the database. 2. Create database connection and implement queries 3. Connect JSON returned by routes to the front end

Owner	Camilo Rivera
Sprint Assigned	4

User Story ID	#130
User Story Name	Categorizer Node
Description	As an actor on the system I wish to see which words in any given string of input belong to any given number of categories (specifically AWL, LO, MED, and HI), if any. I would like it to be presented to me in a simple, easy to use, and color coded fashion.
Acceptance Criteria	<ol style="list-style-type: none">1. Categorize.php – Submit input into categorizer2. Categorizer.php – Categorize arbitrary input into SQL categories3. Select any four categories4. Highlight all matches with selected colors5. Provide a link to the definition of all words, matched or not.
Owner	Charles
Sprint Assigned	3,4,5

User Story ID	#136
User Story Name	Home Page
Description	As a user, I want to access the home-page of the web app so that I can choose which types of words I wish to see.
Acceptance Criteria	<ol style="list-style-type: none">1. Stylize the GUI2. Create buttons3. Set up links to buttons with html and AngularJS code
Owner	Camilo Rivera
Sprint Assigned	5

User Story ID	#120
User Story Name	Words by Category
Description	As a user I want to see the different categorize which the words in my submitted text belong to.
Acceptance Criteria	<ol style="list-style-type: none">1. Create front-end functionality2. Create logic part to compare to words in database3. Return the different words in different categories.
Owner	Camilo Rivera, Charles Benitez

Sprint Assigned	5
-----------------	---

User Story ID	#125
User Story Name	Upload/Scan PDF File
Description	As a user I want to see the words and their different category from a PDF text file so that I may know more about the words in the file.
Acceptance Criteria	<ol style="list-style-type: none"> 1. Allow upload of files 2. Scan text in the pdf file and compare it with words in the database 3. Create an enhanced text of the pdf file.
Owner	Camilo Rivera
Sprint Assigned	6

User Story ID	#121
User Story Name	Enhanced Text
Description	As a user I want to see the words and their different category which the words belong to while reading the full original text.
Acceptance Criteria	<ol style="list-style-type: none"> 1. Display user input with color-coded words by category 2. Compare words with database lists 3. Output whole text.
Owner	Camilo Rivera, Charles Benitez

Sprint Assigned	7
-----------------	---

Pending User Stories:

User Story ID	#132
User Story Name	OCR
Description	As a user I want to see the words and their different category from an image file so that I may know more about the words in the image.
Acceptance Criteria	<ol style="list-style-type: none"> 1. Create front-end functionality 2. Create logic part to compare to words in database 3. Return the different words in different categories.
Owner	Camilo Rivera, Charles Benitez
Sprint Assigned	N/A

User Story ID	#135
User Story Name	Convert GUI to APK / IOS
Description	As a user I want to be able to use this application as a mobile phone app instead of a web app.
Acceptance Criteria	<ol style="list-style-type: none"> 1. Learn conversion libraries. 2. Port to Android 3. Port to IOS
Owner	Camilo Rivera, Charles Benitez

Sprint Assigned	N/A
-----------------	-----

PROJECT PLAN

This section describes the planning that went into the realization of this project. This project incorporated the agile development techniques and as such required the sprints to be planned. These sprint plannings are detailed in the section. This section also describes the components, both software and hardware, chosen for this project.

Hardware and Software Resources

The following is a list of all hardware and software resources that were used in this project:

- Personal laptop computers
- FIU School of Computer Science lab computers
- GoDaddy Linux servers

- PHP5
- MySQL
- AngularJS
- NodeJS
- ExpressJS
- BootStrap CSS
- XAMPP
- MAMP
- Adobe Dreamweaver
- MySQL Workbench, PHPMyAdmin

Sprints Plan:**Sprint planning #1,2:**

Attendees: **Seyedjafar Ehsanzadehsorati, Camilo Rivera, Charles Benitez, Eric Dwyer**

Start time: 4:00pm

End time: 5:00pm

After discussion, the velocity of the team was estimated to be normal.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story #117 MySQL Database
- User Story #123 Landing Page GUI

The team members indicated their willingness to work on the following user stories.

- **Camilo Rivera**
 - User Story #117 MySQL Database
 - User Story #123 Landing Page GUI
- **Charles Benitez**
 - User Story #117 MySQL Database
 - User Story #123 Landing Page GUI

Sprint planning #3:

Attendees: **Seyedjafar Ehsanzadehsorati, Camilo Rivera, Charles Benitez, Eric Dwyer**

Start time: 6:00pm

End time: 7:00pm

After discussion, the velocity of the team was estimated to be normal.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story #123 Landing Page GUI

The team members indicated their willingness to work on the following user stories.

- **Camilo Rivera**
 - User Story #124 K1 words page
 - User Story #123 Landing Page GUI
- **Charles Benitez**
 - User Story #117 MySQL Database
 - User Story #124 K1 words page

Sprint planning #4:

Attendees: **Seyedjafar Ehsanzadehsorati, Camilo Rivera, Charles Benitez, Eric Dwyer**

Start time: 6:00pm

End time: 7:00pm

After discussion, the velocity of the team was estimated to be normal.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story #128 Interface MySQL to Express/Node
- User Story #129 VIRS Admin page
- User Story #126 AWL Page

The team members indicated their willingness to work on the following user stories.

- **Camilo Rivera**
 - User Story #128 Interface MySQL to Express/Node
 - User Story #126 AWL Page
- **Charles Benitez**
 - User Story #129 VIRS Admin page
 - User Story #128 Interface MySQL to Express/Node

Sprint planning #5:

Attendees: **Seyedjafar Ehsanzadehsorati, Camilo Rivera, Charles Benitez, Eric Dwyer**

Start time: 6:00pm

End time: 7:00pm

After discussion, the velocity of the team was estimated to be normal.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story #120 Words by category(text scan)
- User Story #130 Categorizer Node
- User Story #131 Dictionary
- User Story #136 Home Page

The team members indicated their willingness to work on the following user stories.

- **Camilo Rivera**
 - User Story #120 Words by category(text scan)
 - User Story #136 Home Page
- **Charles Benitez**
 - User Story #131 Dictionary
 - User Story #130 Categorizer Node

Sprint planning #6:

Attendees: **Seyedjafar Ehsanzadehsorati, Camilo Rivera, Charles Benitez, Eric Dwyer**

Start time: 6:00pm

End time: 7:00pm

After discussion, the velocity of the team was estimated to be normal.

The product owner chose the following user stories to be done during the next sprint. They are ordered based on their priority.

- User Story #125 Upload/Scan PDF file
- User Story #133 Advanced SQL (uploading)

The team members indicated their willingness to work on the following user stories.

- **Camilo Rivera**
 - User Story #125 Upload/Scan PDF file
- **Charles Benitez**
 - User Story #133 Advanced SQL (uploading)

SYSTEM DESIGN

This section contains information on the design decisions that went into this project. The architecture patterns are outlined and explained. The entire system is shown in a package diagram and the subsystems are explained. Finally, the design patterns used in the project are discussed.

Architectural Patterns

System and Subsystem Decomposition

The architecture being applied is MVC, thus the 3 main subsystems being used are:

Model Subsystem

View Subsystem

Controller Subsystem

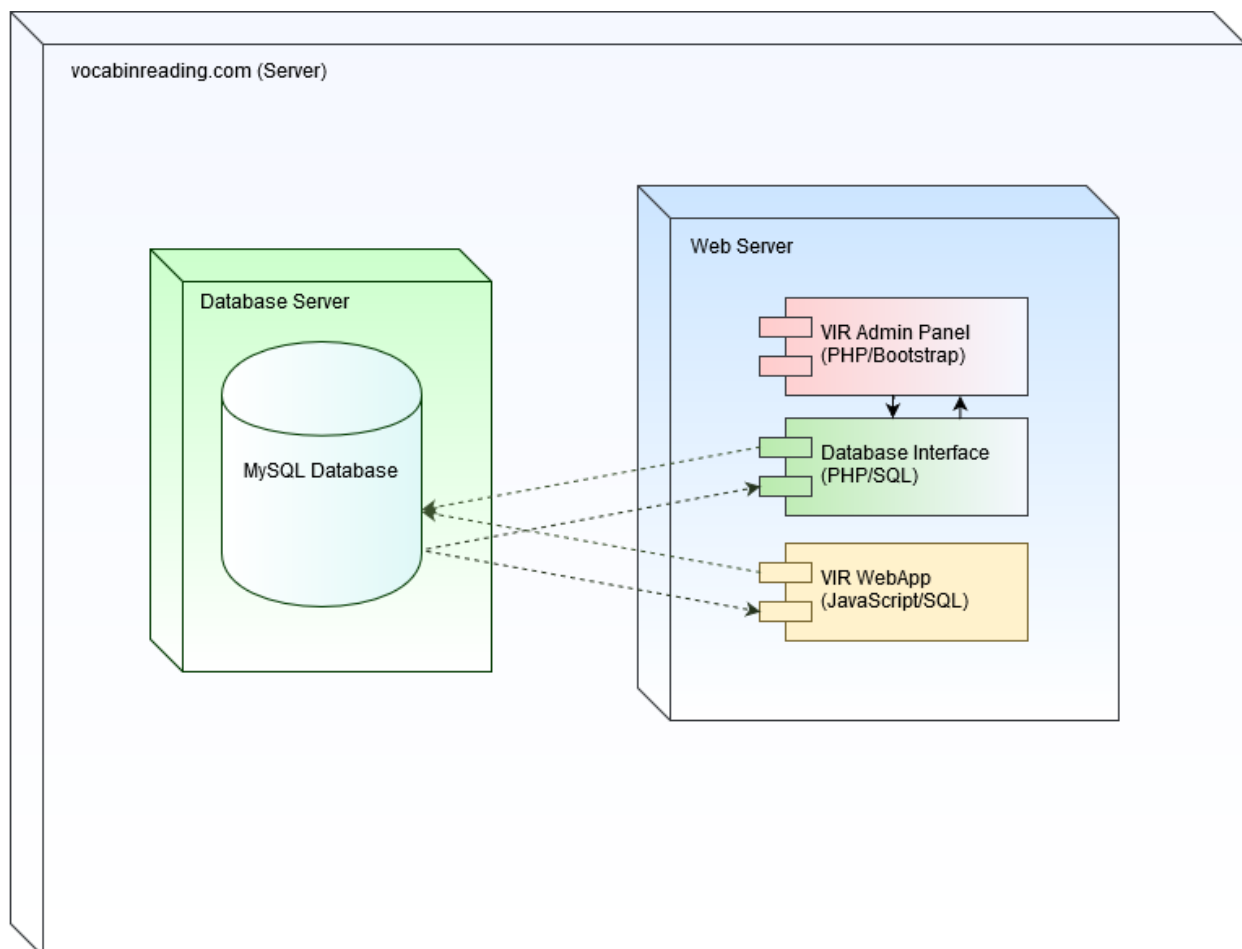
The **View** Subsystem contains all of the code related to the front end of VIR application and all of the graphical components required for interaction with the user. Each of the objects in this subsystem will have a viewable graphical interface that will either display some information to the users or take inputs and pass them on to another subsystem which will perform some operation with those inputs. Most use cases are involved with the interface since the user must make use of this subsystem to be able to perform tasks. Most of the code here consists of Html/CSS and AngularJS injected for functionality.

The **Controller** Subsystem is responsible for controlling and processing that information given by input. It then performs some task assign by the system. This subsystem consists of a set of controllers that will contain methods that decide through set of conditions on how some input accepted by the view will be handled and executed. This subsystem handles most of the use cases being implemented since all the logic of the program will be handled here. All interface interactions done by the user in the view will be taken to the Controller Subsystem and have

tasks performed on it. Most of the work here is done with JavaScript through application of AngularJS framework.

The **Model** Subsystem only deals with storing, sending and retrieving information from a MySQL database that will hold the objects required for the application to function properly. This subsystem is mainly composed of objects that will either read or update the database. This system handles tasks required by the MySQL database. ExpressJS is used to route the data from and to the database and MySQL is used to store and organize it.

Deployment Diagram

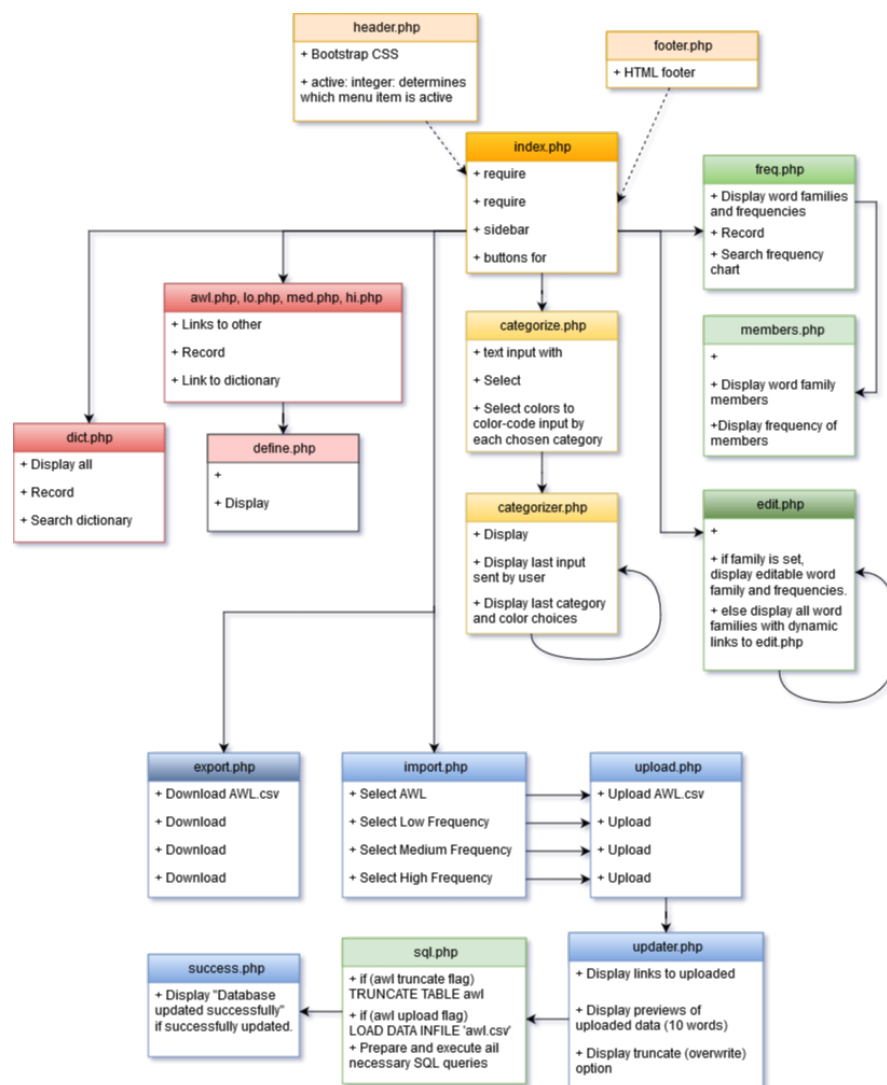


Design Patterns:

Some of the design patterns implemented in the VIR web app are:

- Factory design pattern for data transfer between controller and routes.
- Singleton for the Angular App.

The design for the VIR Admin Panel is shown in the figure below:



SYSTEM VALIDATION:

Test cases performed

Test case ID: 117a

Description/Summary of Test: Does the MySQL database work

Pre-condition: Data Imported

Expected Results: Yes

Actual Result: Yes

Status (Fail/Pass): Pass

Test case ID: 117b

Description/Summary of Test: Does the administrative panel work

Pre-condition: Data Imported

Expected Results: Yes

Actual Result: Yes

Status (Fail/Pass): Pass

Test case ID: 123a

Description/Summary of Test: Does the landing page work

Pre-condition: Source code completed

Expected Results: Yes

Actual Result: Yes

Status (Fail/Pass): Pass

Test case ID: 123b

Description/Summary of Test: Does the link to K1 work

Pre-condition: Source code completed

Expected Results: Yes

Actual Result: Yes

Status (Fail/Pass): Pass

•

Test case ID: 128a

Description/Summary of Test: Do queries work

Pre-condition: Source code completed

Expected Results: Yes

Actual Result: Yes

Status (Fail/Pass): Pass

Test case ID: 128b

Description/Summary of Test: Do words in database print?

Pre-condition: Source code completed

Expected Results: Yes

Actual Result: Yes

Status (Fail/Pass): Pass

Test case ID: 120a

Description/Summary of Test: Does input work

Pre-condition: Source code completed

Expected Results: Yes

Actual Result: Yes

Status (Fail/Pass): Pass

Test case ID: 120b

Description/Summary of Test: Does word comparison work

Pre-condition: Source code completed

Expected Results: Yes

Actual Result: Yes

Status (Fail/Pass): Pass

GLOSSARY:

1. Analysis: detailed examination of the elements or structure of something, typically as a basis for discussion or interpretation.
2. MySQL: Relational database used for the app
3. AngularJS: A JavaScript framework used for developing front-end and controller work.
4. ExpressJS: A JavaScript framework used for developing back-end and routing.
5. NodeJS: A JavaScript framework used for server functionality in running the web app.
6. Categorize: place in a particular class or group.
7. GUI: Graphical user interface
8. Subsystem: a self-contained system within a larger system.
9. Panel: a predefined display image on a user interface.

APPENDIX

Appendix A - UML Diagrams

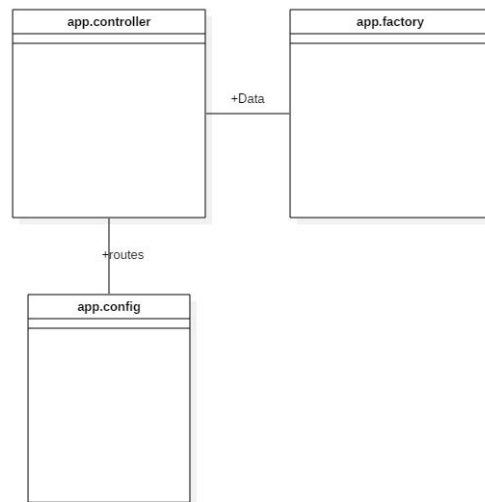
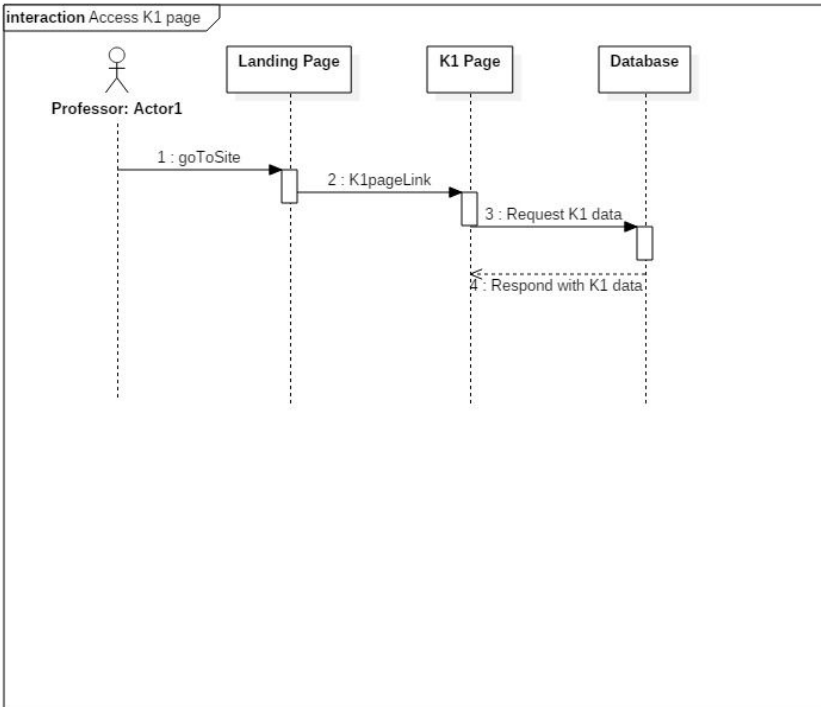
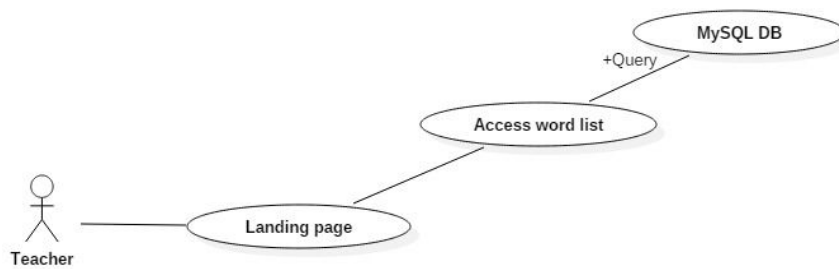
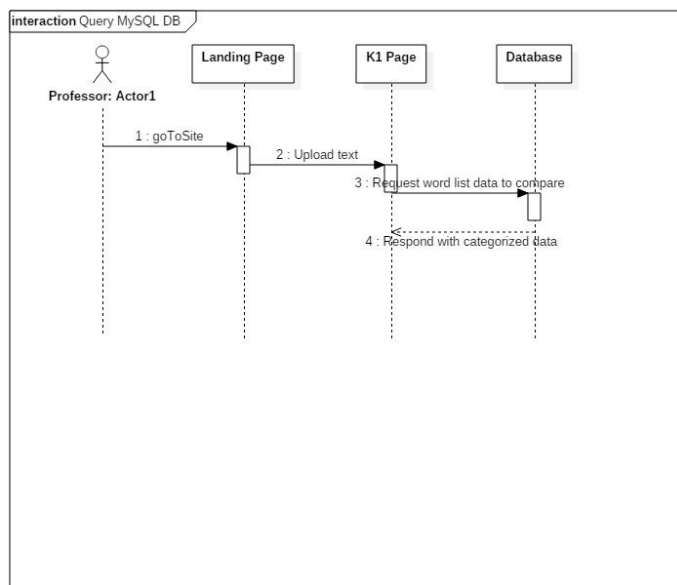
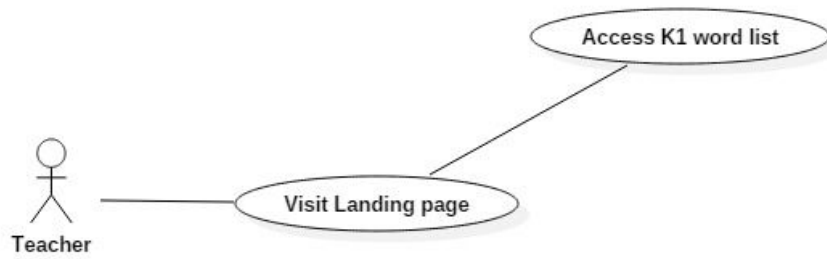
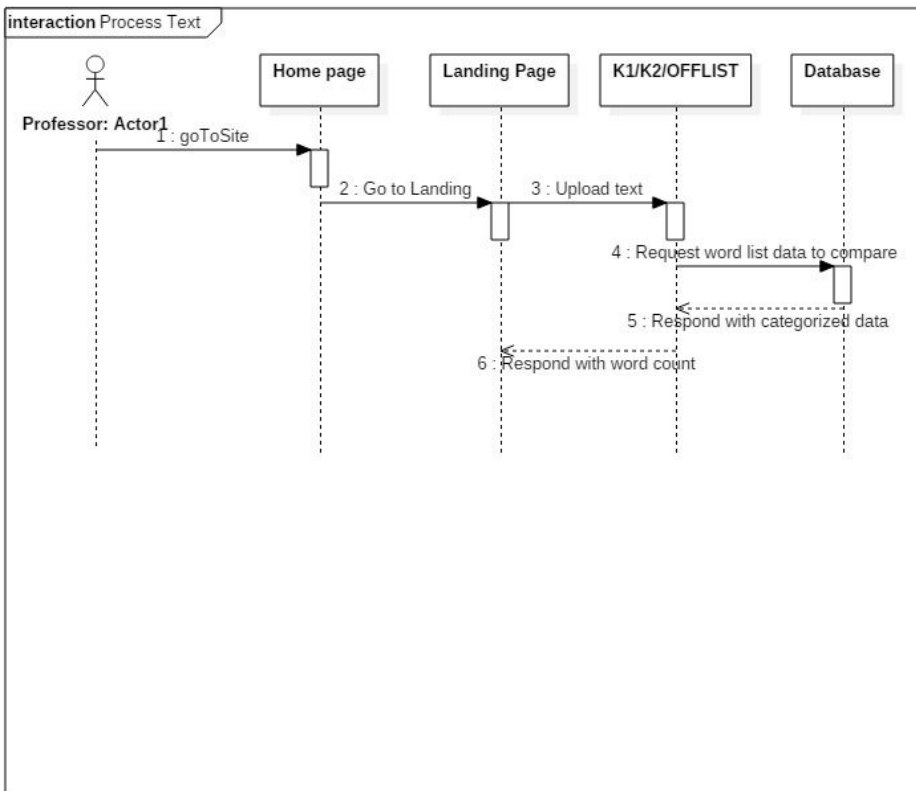
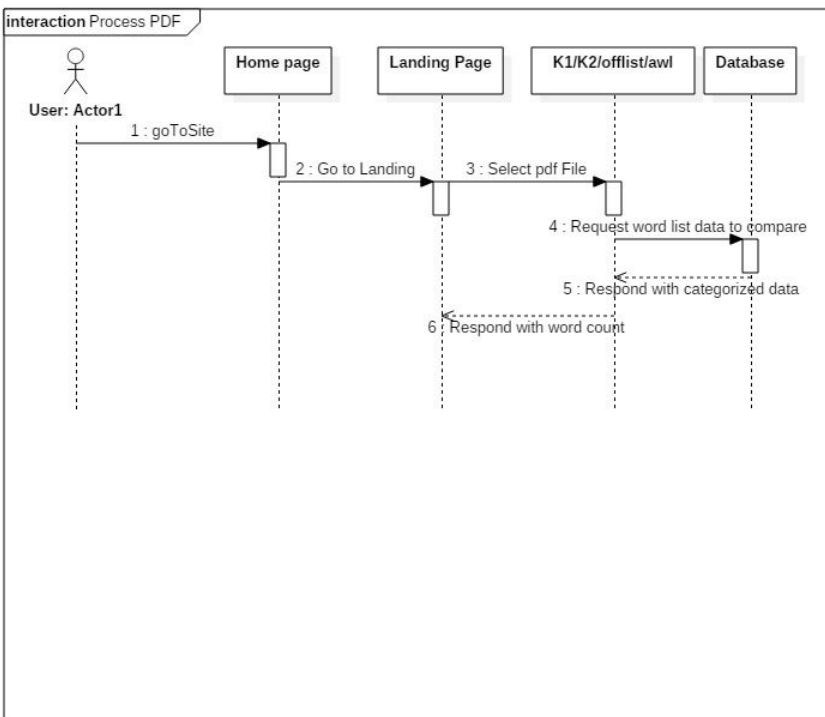
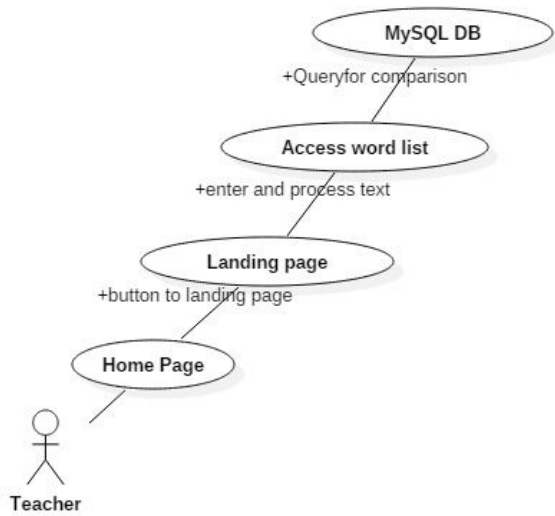


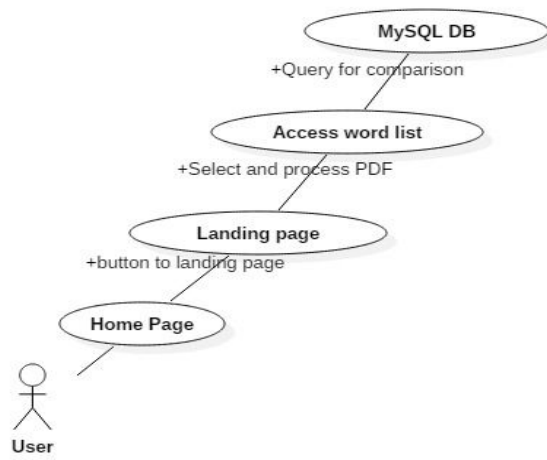
Figure 1: Angular Controller subsystem design











Appendix B - User Interface Design

Type your text

Enter Text

Process Text

Insert Image File

JPG

File:

Process Text

Insert PDF File

PDF

File:

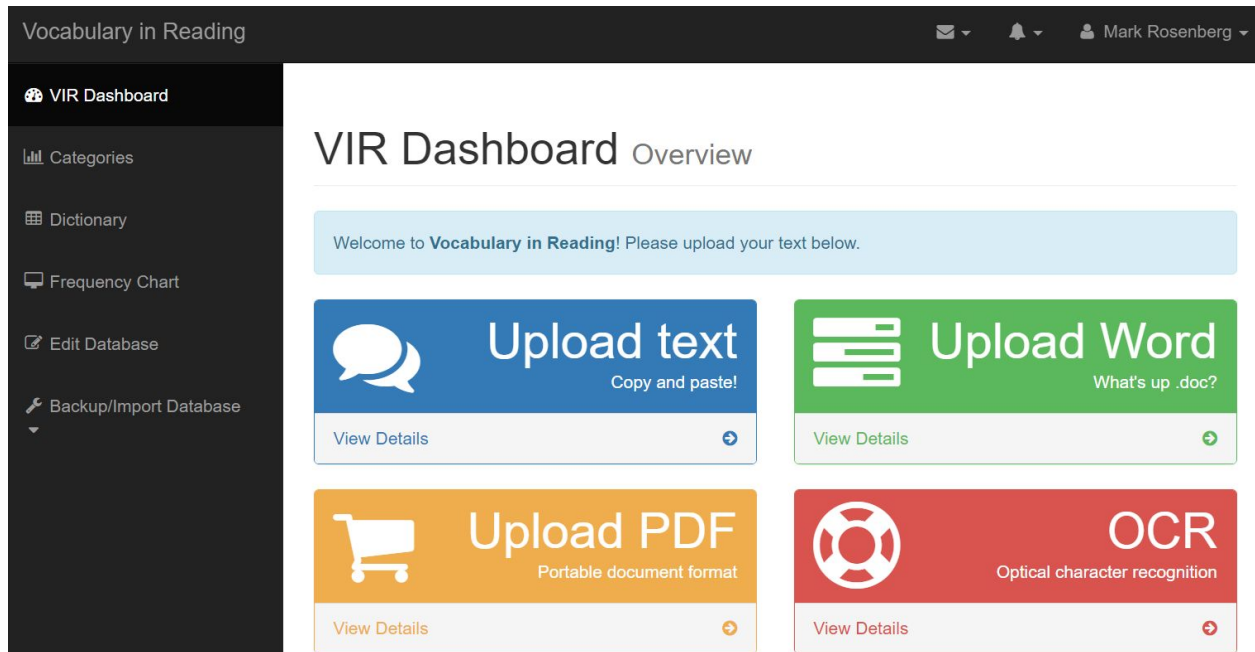
Process Text

Number of words	
High frequency	0
Medium Frequency	0
Low Frequency	0
Academic Word List	0


Proceed to Enhanced Text

Web App GUI design

VIR Admin Panel Screenshots:



Vocabulary in Reading

 VIR Dashboard

Categories

Dictionary

Frequency Chart

Edit Database

Backup/Import Database

Output:

A secret always has a strengthening effect upon a newborn friendship as does the shared impression than an external figure is to blame the men of the Crown have become united less by their shared beliefs we observe than by their shared misgivings—which are in the main externally directed In their analyses variously made of Alastair Lauderback George Shepard Lydia Wells Francis Carver Anna Wetherell and Emery Staines the Crown men have become more and more suggestive despite the fact that nothing has been proven no body has been tried and no new information has come to light Their beliefs have become more fanciful their hypotheses less practical their counsel less germane Unconfirmed suspicion tends over time to become wilful fallacious and prey to the vicissitudes of mood—it acquires all the qualities of common superstition—and the men of the Crown Hotel whose nexus of allegiance is stitched after all in the bright thread of time and motion have like all men no immunity to influence

Input:

A secret always has a strengthening effect upon a newborn friendship, as does the shared impression than an external figure is to blame: the men of the Crown have become united less by their shared beliefs, we observe, than by their shared misgivings—which are, in the main, externally directed. In their analyses, variously made, of Alastair Lauderback, George Shepard, Lydia Wells, Francis Carver, Anna Wetherell, and Emery Staines, the Crown men have become more and more suggestive, despite the fact that nothing has been proven, no body has been tried, and no new information has come to light. Their beliefs have become more fanciful, their hypotheses less practical, their counsel less germane. Unconfirmed suspicion tends, over time, to become wilful, fallacious, and prey to the vicissitudes of mood—it acquires all the qualities of common superstition—and the men of the Crown Hotel, whose nexus of allegiance is stitched, after all, in the bright thread of time and motion, have, like all men, no immunity to influence.

☒ AWL

Lime

☒ Low Frequency

Blue


☒ Medium Frequency


Red


☒ High Frequency

Gold

Categorize





 Mark Rosenberg

4/18/2017

Page 34 of 38

The screenshot displays the 'Vocabulary in Reading' application interface. The top navigation bar includes the application name, a mail icon, a bell icon, and the user name 'Mark Rosenberg'. The left sidebar contains a menu with the following items: 'VIR Dashboard', 'Categories' (selected), 'Dictionary', 'Frequency Chart', 'Edit Database', and 'Backup/Import Database'. The main content area is titled 'Category Viewer' and shows the 'Academic word list - Low - Medium - High'. A status bar indicates 'Showing words 1 to 50 of 1390 total words.' Below this, there are navigation links: 'First page', 'Previous page', 'Next page', and 'Last page'. A list of words is displayed, including 'abandon', 'abandonment', 'abnormal', 'abnormally', 'abstract', 'abstraction' (highlighted), 'abstractly', and 'academia'. A modal window titled 'Untitled Document - Google Chrome' is overlaid on the bottom right, showing the definition for the word 'Abstraction' from a local file: 'Definition for word *Abstraction* a. "The act of abstracting, separating, or withdrawing, or the state of being withdrawn; withdrawal."'. The modal also includes 'Go back' and 'Close' links.

Vocabulary in Reading

VIR Dashboard

Categories

Dictionary

Frequency Chart

Edit Database

Backup/Import Database

Category Viewer

Academic word list - Low - Medium - High

Showing words 1 to 50 of 1390 total words.

First page - Previous page - Next page - Last page

abandon

abandonment

abnormal

abnormally

abstract

abstraction

abstractly

academia

accommodation

Untitled Document - Google Chrome

localhost/admin/define.php?word=abstraction

Definition for word *Abstraction* a.

"The act of abstracting, separating, or withdrawing, or the state of being withdrawn; withdrawal."

[Go back](#) [Close](#)

Vocabulary in Reading

VIR Dashboard

Categories

Dictionary

Frequency Chart

Edit Database

Backup/Import Database

Import Data

Export Data

Backup/Import Database

Import Data

Backup/Import Database

Import Data

The following files have been uploaded to the server. If no files are shown, the files were not accepted by the system. In that case, please [try again](#).

WARNING: It is **YOUR** responsibility to check that the files are correct before attempting to update the database, including checking if the columns are correct. The previews **WILL** be the data going into the database. Good luck.

Hint: Selecting **TRUNCATE** will **delete all existing data in the database** for that specific list before adding the uploaded data. Not selecting it will append the uploaded data to the end of the existing data.

AWL File Uploaded : AWL.csv

TRUNCATE before updating AWL ? ☒

Previewing AWL table . . .

test0
test1
test2
test3
test4
test5
test6
test7
test8
test9

Update SQL Database

[Back](#)

Appendix C - Sprint Review Reports

Appendix D - User Manuals, Installation/Maintenance Document, Shortcomings/Wishlist Document and other documents

User Manual Web App:

1. First enter the domain URL.
2. Once you enter the URL you are the home page which does not contain functionality but hold the gate into the landing page.
3. Click the button that says Proceed to Analyze to continue to the landing page
4. In the landing page:
 - a. You have 1 textbox for input and 2 file input options
 - b. You have 3 Process Text buttons each belonging to a input choice:
 - i. Text
 - ii. PDF File
 - iii. Image File
 - c. Below you have 5 buttons
 - i. Button for High Frequency
 - ii. Button for Medium Frequency
 - iii. Button for Low Frequency
 - iv. Button for Proceed to Enhanced Text
 - d. Before getting any data from these buttons, you must first enter any of the 3 types of input and then hit Process Text button belonging to that input.
 - e. After hitting entering an input and hitting Process Text, the number of words belonging to each category will be displayed next to the buttons representing each category.
 - f. You will now get data belonging to each word in their corresponding categories. It will show you the word and the definition.
 - g. If you go to Enhanced text, it will show you the entire original text but color coded with each color representing one of the four categories above.
 - h. If you wish to reset all the data, simply hit refresh.

REFERENCES:

Resources used to help develop project:

For uploading files in Angular:

<https://github.com/danialfarid/ng-file-upload>

For reading PDF files:

<https://github.com/mozilla/pdf.js>

For learning and implementing MEAN stack:

<https://thinkster.io/>

For past MEAN projects for reference:

<https://github.com/crive150>