*Florida International University*

*School of Computing and Information Science*

Final Deliverable

Project:

Vocabulary In Reading Study

Current Team Members: Diego Chacin, Regine Donatien, Javier Duenaz, Jason Naya, Michelle Cruz

Product Owners: Seyedjafar Ehsanzadehsorati

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

*This document goes over the Vocabulary in Reading Study mission and the software solutions that were implemented. The purpose of Vocabulary in Reading Study is to provide english language learners and educators with a tool that will allow them to break down the difficulty of any given word in a text as well as the text itself. The main focus of our work was in optimizing the user experience of the website, optimizing the image recognition function of the website, and adding additional functionality to the user storage and login system.*

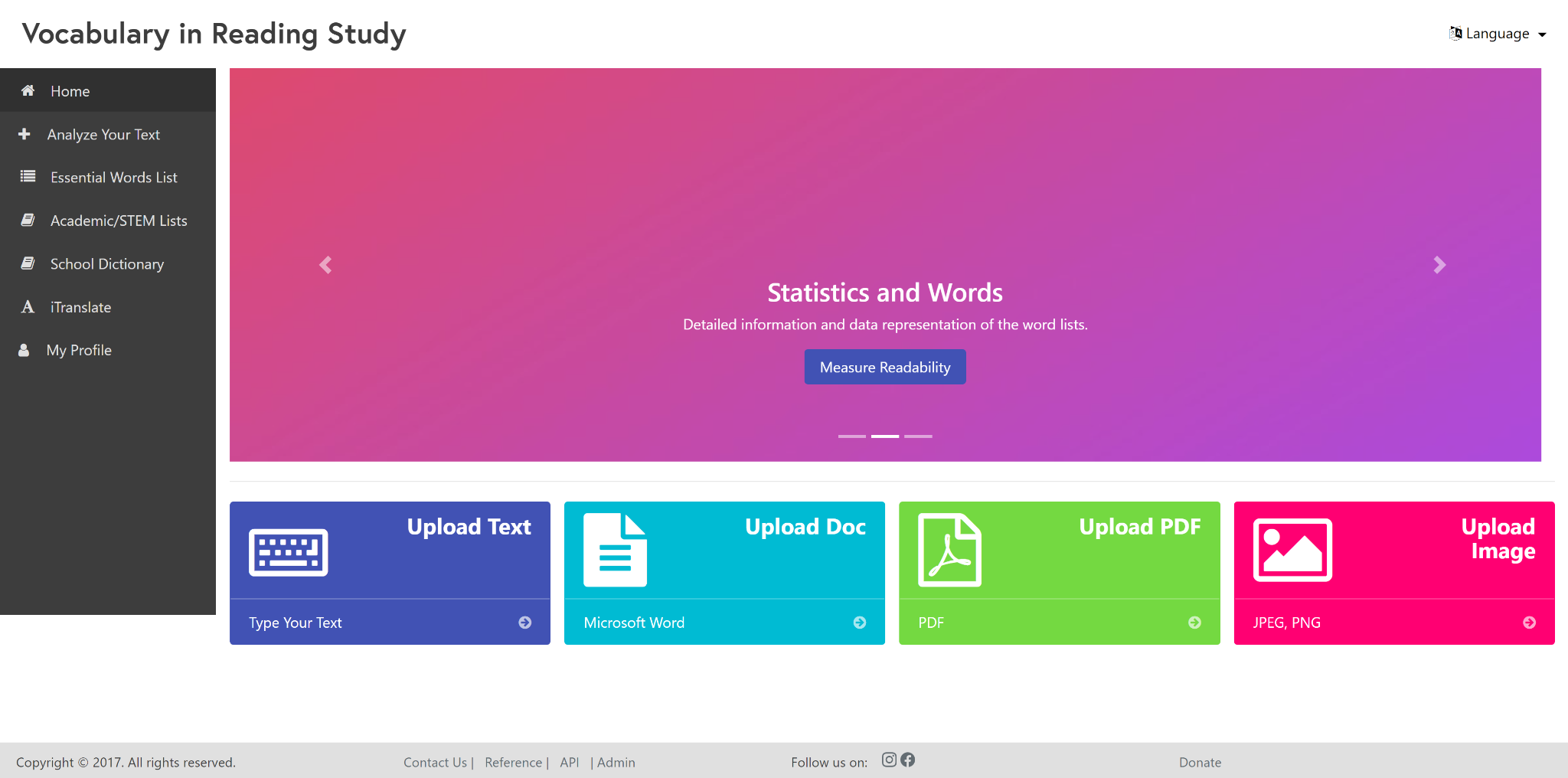
# Introduction

## Current System

This is the seventh iteration of the application. The current system is hosted on AWS (Amazon

Web Servers) on an on-demand EC2 instance. It uses AWS RDS with MySQL for the database,

Spring MVC, and Angular/Typescript. Currently the system can analyze text from either typed text submitted through a form or by uploading a file. The website can support the following file extensions: .docx, .pdf, .png and .jpeg. The site gives the user the option to “enhance text” which shows the difficulty of the words found in the text and displays dictionary definitions of the words once you click on them. There is also the statistics feature that gives an in-depth analysis of the text. There is a translator that translates text to many languages. Users can also create profiles as well as test their vocabulary with tests to assess their progression in learning English. There are administrative tools for the product owners to add and remove words from the dictionary/essential words list. The application now has an updated UI throughout the site with improved text recognition for images.



## Purpose of the New System

This semester (VIRS version 7.0), we improved upon some areas of the application: We improved the text recognition for uploaded images and formatted the output in a way that is easy to read. We also updated the UI to better suit current design practices. More notably, the mobile version of the site is now more adaptable for smaller viewports and formatting errors have been removed. Users now have the ability to download the words lists on all the list pages. As per the product owner’s request the vocabulary tests tab has been removed, and a button has been added to cite the website. We also worked on creating a mobile application for the website for iOS devices.

# User Stories

## Implemented User Stories

**User Story [#1]** - Run Coding Environment (10 points)

**Title:** As a software developer, I would like to run the project environment on my personal computer in order to make the necessary changes and improvements to the website.

**Acceptance Criteria:**

Acceptance Criteria 1: Developer can view the front-end on localhost:4200

Acceptance Criteria 2: Developer can successfully build the backend user virs.cmd with no errors

Acceptance Criteria 3: Developer can view the entire application on localhost:8080

Acceptance Criteria 4: Developer can make edits and have the application reflect those changes on localhost:8080

**Estimate:** 5\* 4 = 20 points

**User Story [#2]** - Deploy to AWS

**Title:** As a developer I want to figure out how to deploy new builds to AWS so that we can work

on the project.

**Acceptance Criteria:**

Acceptance Criteria 1: Make sure we keep backups

Acceptance Criteria 2: Make sure that changes are reflected on myvirs.com

Acceptance Criteria 3: Verify Elastic Beanstalk accepts the file

**Estimate**: 5\* 4 = 20 points

**User Story [#3]** - Retrieve Vocabulary from PDF, Images, DOCX and text

**Title:** As a user, I want to be able to upload .pdf, .png, .docx, .jpeg file to the web application in order to

retrieve the vocabulary.

**Acceptance Criteria:**

Acceptance Criteria 1: Users are able to upload a .pdf document with no errors.

Acceptance Criteria 2: Users are able to upload a .png document with no errors.

Acceptance Criteria 3: Users are able to upload a .docx document with no errors.

Acceptance Criteria 4: Users are able to upload a .jpg document with no errors.

Acceptance Criteria 5: Users are able to text into a form with no errors.

Acceptance Criteria 6: The words are pulled from the .pdf document and are correctly displayed, categorized and color coded.

Acceptance Criteria 7: The words are pulled from the .png document and are correctly displayed, categorized and color coded.

Acceptance Criteria 8: The words are pulled from the .docx document and are correctly displayed, categorized and color coded.

Acceptance Criteria 9: The words are pulled from the .jpg document and are correctly displayed, categorized and color coded.

Acceptance Criteria 6: The words are pulled from the text form and are correctly displayed, categorized and color coded.

**Estimate:** 20 points

**User Story [#4] -** Cite Word Lists

**Title:** As a user I want to cite the word list pages

**Acceptance Criteria:**

Acceptance Criteria 1: The cite us button copies citation to the user’s clipboard

**Estimate:** 3 points

**User Story [#5]** - Fix Image Recognition Software

**Title:** As a user, I would like to upload an image to the website and have it recognize the text accurately.

**Acceptance Criteria:**

Acceptance Criteria 1: The image is processed.

Acceptance Criteria 2: All the text from the image is displayed.

Acceptance Criteria 3: All the text from the image is in the correct order

**Estimate:** 10 points

**User Story [#6]** - Remove Frequency Column

**Title:** As a product owner, I would like to have the frequency column removed since it fluctuates often and does not add value to the site.

**Acceptance Criteria:**

Acceptance Criteria 1: The user no longer sees the frequency column on any of the word lists.

**Estimate:** 2 points

**User Story [#7]** - Update the Word Lists

**Title:** As a product owner, I would like to update the word lists so it reflects current vocabulary trends.

**Acceptance Criteria:**

Acceptance Criteria 1: The product owner can login into the admin portal

Acceptance Criteria 2: The new word is reflect on the website

**Estimate:** 5 points

**User Story [#8]** - Update/Refine UI

**Title:** As a software developer, I would like to make the website more user friendly and fix the formatting of the mobile version of the site.

**Acceptance Criteria:**

Acceptance Criteria 1: All text is visible on desktop. No elements are obscured by another.

Acceptance Criteria 2: The word list columns are evenly spaced and formatted correctly

Acceptance Criteria 3: The footer is consolidated and takes up less real estate on the screen

Acceptance Criteria 4: Reduce the number of different colors on each page

Acceptance Criteria 5: All text is visible on mobile. No elements are obscured by another.

Acceptance Criteria 6: The carousel is removed from mobile and replaced by a still image

Acceptance Criteria 8: The icons on the footer match thematically.

**Estimate:** 10 points

**User Story [#10] -** Download Word Lists

**Title:** As a user, I would like to download the word lists on the website as an Excel Spreadsheet.

**Acceptance Criteria:**

Acceptance Criteria 1: The “Download List” button converts the displayed table into an .xlsx

Acceptance Criteria 2: The .xlsx is exported to the user’s device

**Estimate:** 5 points

**User Story [#11]**- Dialog box that explains the readability score

**Title:** As a user, I would like to know more information about the readability score that appears on the Statistics page

**Acceptance Criteria:**

Acceptance Criteria 1: The button is visible on the Statistics page

Acceptance Criteria 2: The button once clicked shows the explanation on the different scores

Acceptance Criteria 3: The dialog box can be closed

**Estimate:** 2 points

**User Story [#12] -** Research Image Processing implementation

**Title:** As a user, I would like to understand the back end of image processing.

**Acceptance Criteria:**

Acceptance Criteria 1: Each section of image processing is well understood with context to one another

Acceptance Criteria 2: The .xlsx is exported to the user’s device

**Estimate:** 2 points

**User Story [#13] -** Research Image Processing implementation

**Title:** As a user, I would like to understand the back end of image processing.

**Acceptance Criteria:**

Acceptance Criteria 1: Each section of image processing is well understood with context to one another

Acceptance Criteria 2: The .xlsx is exported to the user’s device

**Estimate:** 2 points

**User Story [#14] -** Research Block Recognition Code

**Title:** As a user, I would like to be able to upload an image of any format.

**Acceptance Criteria:**

Acceptance Criteria 1: We properly understand how the block recognition and column code works

**Estimate:** 2 points

**User Story [#15] -** Change margin of error and clean up column recognition code

**Title:** As a user, I would like to be able to upload an image of any quality.

**Acceptance Criteria:**

Acceptance Criteria 1: We properly fix the bugs in the column recognition code.

**Estimate:** 10 points

**User Story [#16] -** Research image optimization

**Title:** As a user, I would like to be able to upload an image of any quality without compromising output.

**Acceptance Criteria:**

Acceptance Criteria 1: We properly understand how to optimize an image for a block reader to function properly

**Estimate:** 2 points

**User Story [#17] -** Implement image optimization code

**Title:** As a user, I would like to be able to upload an image of any quality.

**Acceptance Criteria:**

Acceptance Criteria 1: An image of any quality can be inputted to the textract system without losing any blocks

**Estimate:** 7 points

**User Story [#18] -** research into iOS app

**Title:** As a user, I would like to be able to be able to use an app for VIRS.

**Acceptance Criteria:**

Acceptance Criteria 1: We understand how to make an iOS application.

**Estimate:** 5 points

**User Story [#19] -** Download all tools needed for iOS app

**Title:** As a user, I would like to be able to be able to use an app for VIRS.

**Acceptance Criteria:**

Acceptance Criteria 1: All tools needed to create the iOS app is downloaded.

**Estimate:** 2 points

**User Story [#20] -** Learn how to use XCode storyboards and Swift 5

**Title:** As a user, I would like to be able to be able to use an app for VIRS.

**Acceptance Criteria:**

Acceptance Criteria 1: sufficiently comfortable using XCode and swift 5 programming language.

**Estimate:** 5 points

**User Story [#21] -** Begin coding the iOS app

**Title:** As a user, I would like to be able to be able to use an app for VIRS.

**Acceptance Criteria:**

Acceptance Criteria 1: iOS app is functioning.

**Estimate:** 10 points

## Pending User Stories

**User Story [#1]**- Update Angular CLI

**Title:** As a software developer, I would like to update the Angular CLI to use all the new features that recent updates have brought to the VIRS site.

**Acceptance Criteria:**

Acceptance Criteria 1: The Angular CLI mismatch warning message no longer appears on the console

**Estimate:** 5 points

**User Story [#2]**- Publish the VIRS application to the Google Play Store

**Title:** As a software developer, I would like to publish the VIRS application to the Google Play Store.

**Acceptance Criteria:**

Acceptance Criteria 1: The VIRS application is approved and published to the Google Play Store.

**Estimate:** 5 points

**User Story [#3]**- Publish the VIRS application to the Apple App Store

**Title:** As a software developer, I would like to publish the VIRS application to the Apple App Store.

**Acceptance Criteria:**

Acceptance Criteria 1: The VIRS application is approved and published to the Apple App Store.

**Estimate:** 5 points

**User Story [#4]**- Secure the VIRS site

**Title:** As a software developer, I would like to secure the website.

**Acceptance Criteria:**

Acceptance Criteria 1: The browser shows https://

**Estimate:** 5 points

# 

# 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:

Note: This list of hardware and software stayed the same throughout this semester, so this

is the list made by the previous team.

### Hardware

* Computer running Linux, Mac OS or windows.
* For the deployed application make sure you stay within the AWS server constraints
* 64bit Amazon Linux 2017.03 v2.5.5 running Java 8

### Software

The following list is the software used in the application. Note that it is quite extensive and includes all the development layers of the stack.

#### Front end

* Angular 4.3.1
* Ng-Bootstrap 1.0.0
* Ng-Translate 7.0.0
* Chat.js 2.7.1
* Font-Awesome 4.7.0
* Ng2-Charts 1.6.0
* Rxjs 5.1.0
* Jasmine 2.5.45
* Karma 1.7
* Typescript 2.3.3

#### Back end

* Maven 3.5
  + Commons-lang3 3.4
  + HikariCP 1.5.6.RELEASE
  + Jai-imageio-core 1.3.1
  + Jai-imageio-jpeg2000 1.3.0
  + Jasypt-spring-boot-starter 1.16
  + Levigo-jbig2-imageio 2.0
  + Mysql-connector-java 1.5.6.RELEASE
  + Opencsv 3.3
  + Opencv 3.2.0-1
  + Spring-boot-starter-data-jpa 1.5.6.RELEASE
  + Spring-boot-starter-security 1.5.6.RELEASE
  + Spring-boot-starter-test 1.5.6.RELEASE
  + Spring-boot-starter-web 1.5.6.RELEASE
  + Springfox-swagger-ui 2.7.0
  + Springfox-swagger2 2.7.0
  + Sqlite-jdbc 1.5.6.RELEASE
  + Tess4j 3.4.1
  + Thucydides-core 0.9.275
  + Tika-parsers 1.16
* Tesseract 3.05.01
* Leptonica 1.74.4
* Mysql 14.14
* Java 1.8.0
* Spring Developer Suite 3.9

#### iOS

* Xcode 12.1 (or latest version of xcode)
* Swift 5 (Storyboards not SwiftUI)
* Command line tools 9.2

#### Other

* Git 2.14.1
* Bash 3.2.57

# Sprints Plan

## Sprint 1

**Sprint Planning Meeting Minutes: 30**

Attendees: Diego Chacin

Regine Donatien

Javier Duenas

Jason Naya

Michelle Cruz

Start time: 7:00 pm

End time: 7:30 pm

Sprint goal: Improve imaging software on the website, finish work on lost username option.

After discussion, the velocity of the team was estimated to be: 25 hours

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

* User Story [#1]: Code lost username option in the login in the HTML, get running on back end. **Point 5 \*5 (Assigned to all group members.)**
* User Story [#2] Port graphics from lost password over to lost username Point 4 (Assigned to all group members)
* User Story [#3] Research imaging software, start to look for and document possible improvements Point 4 (Assigned to all group members)

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

\*\*Indicated next to the user stories.

## Sprint 2

Sprint Planning Meeting Minutes: 30

Attendees: Diego Chacin

Regine Donatien

Javier Duenas

Jason Naya

Michelle Cruz

Start time: 7:00 pm

End time: 7:30 pm

Sprint goal: Improve imaging software on the website, finish work on lost username option.

After discussion, the velocity of the team was estimated to be: 25 hours

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

User Story [#1]: Code lost username option in the login in the HTML, get running on back end. Point 5 \*5 (Assigned to all group members.)

User Story [#2] Port graphics from lost password over to lost username Point 4 (Assigned to all group members)

User Story [#3] Research imaging software, start to look for and document possible improvements Point 4 (Assigned to all group members)

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

\*\*Indicated next to the user stories.

## Sprint 3

**N/A**

## Sprint 4

Sprint Planning Meeting Minutes: 30

Attendees: Diego Chacin

Regine Donatien

Javier Duenas

Jason Naya

Michelle Cruz

Start time: 7:00 pm

End time: 7:30 pm

Sprint goal: Improve imaging software on the website, finish work on lost username option.

After discussion, the velocity of the team was estimated to be: 25 hours

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

User Story [#1] Port graphics from lost password over to lost username Point 4 (Assigned to all group members)

User Story [#2] Research imaging software and fix errors (Assigned to all group members)

User Story [#3] Update UI (Regine Donatien)

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

\*\*Indicated next to the user stories.

## Sprint 5

**Sprint Planning Meeting Minutes: 30**

Attendees: Diego Chacin

Regine Donatien

Javier Duenas

Jason Naya

Michelle Cruz

Start time: 7:00 pm

End time: 7:30 pm

Sprint goal: Improve imaging software on the website, finish work on lost username option.

After discussion, the velocity of the team was estimated to be: 25 hours

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

* User Story [#1] Create a modal explaining readability (Assigned to all group members)
* User Story [#2] Research imaging software and fix errors (Assigned to all group members)
* User Story [#3] Download word lists from website (Assigned to all group members)
* User Story [#4] Create a citation (Assigned to all group members)

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

\*\*Indicated next to the user stories.

## Sprint 6

**Sprint Planning Meeting Minutes: 30**

Attendees: Diego Chacin

Regine Donatien

Javier Duenas

Jason Naya

Michelle Cruz

Start time: 7:00 pm

End time: 7:30 pm

Sprint goal: Improve imaging software on the website, finish work on lost username option.

After discussion, the velocity of the team was estimated to be: 25 hours

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

* User Story [#2] Research imaging software and fix errors (Assigned to all group members)
* User Story [#3] Download word lists from website through backend instead of frontend (Assigned to all group members)

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

\*\*Indicated next to the user stories.

# 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.

Note: Our System Design Section has not changed this semester, so the following System Design info is the same as previous semesters’.

## Architectural Patterns

Model View Controller is the main design for the architecture. We are separating the main three

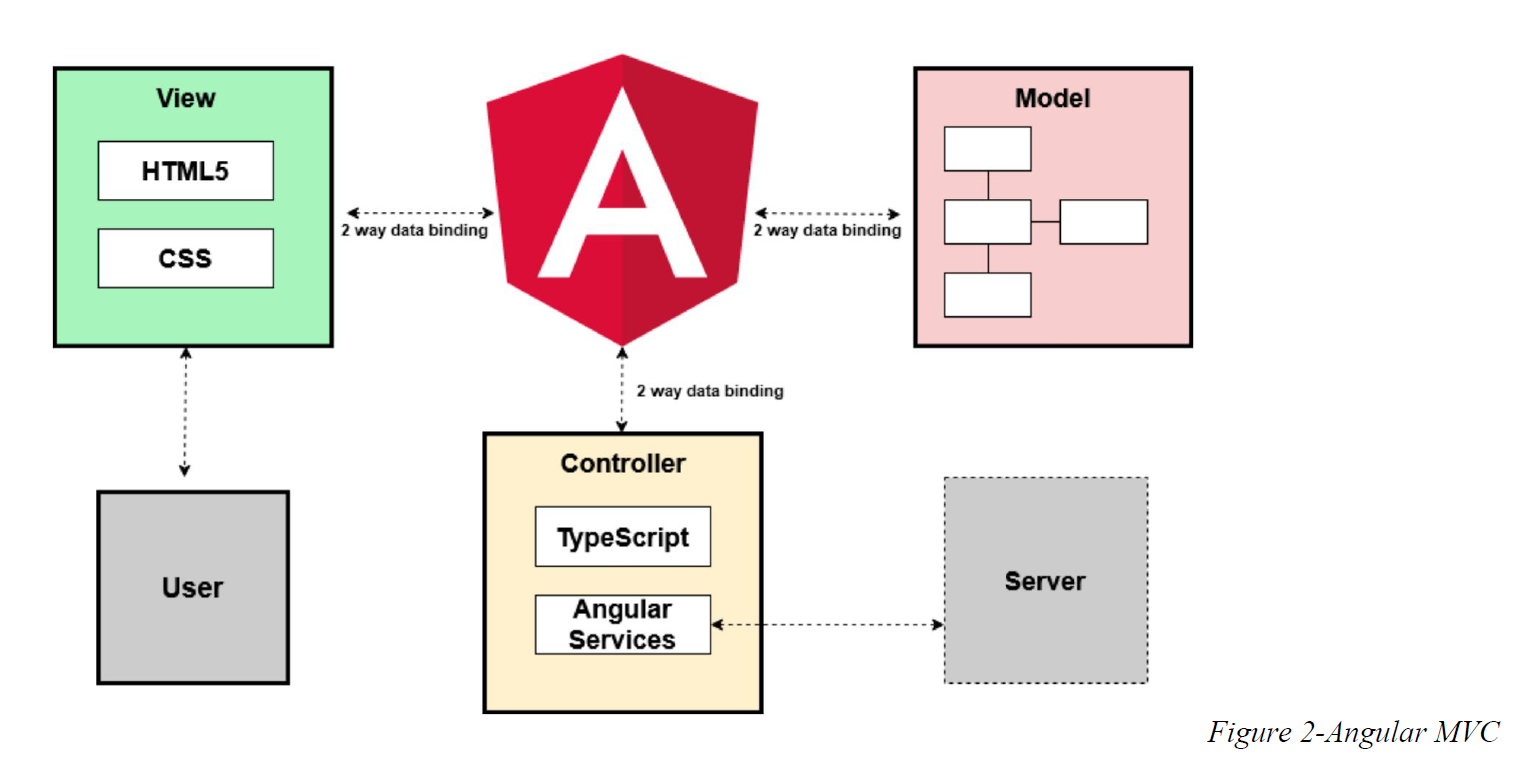
parts of the application: user interaction, processing of information and storage. The segregation

of these concerns favors production since each of the developers work in a separate section. This separation of components makes the system easily modifiable in the future. With this design, we ensure that we have multiple views for a controller; the system produces APIs that can be consumed by any other application.

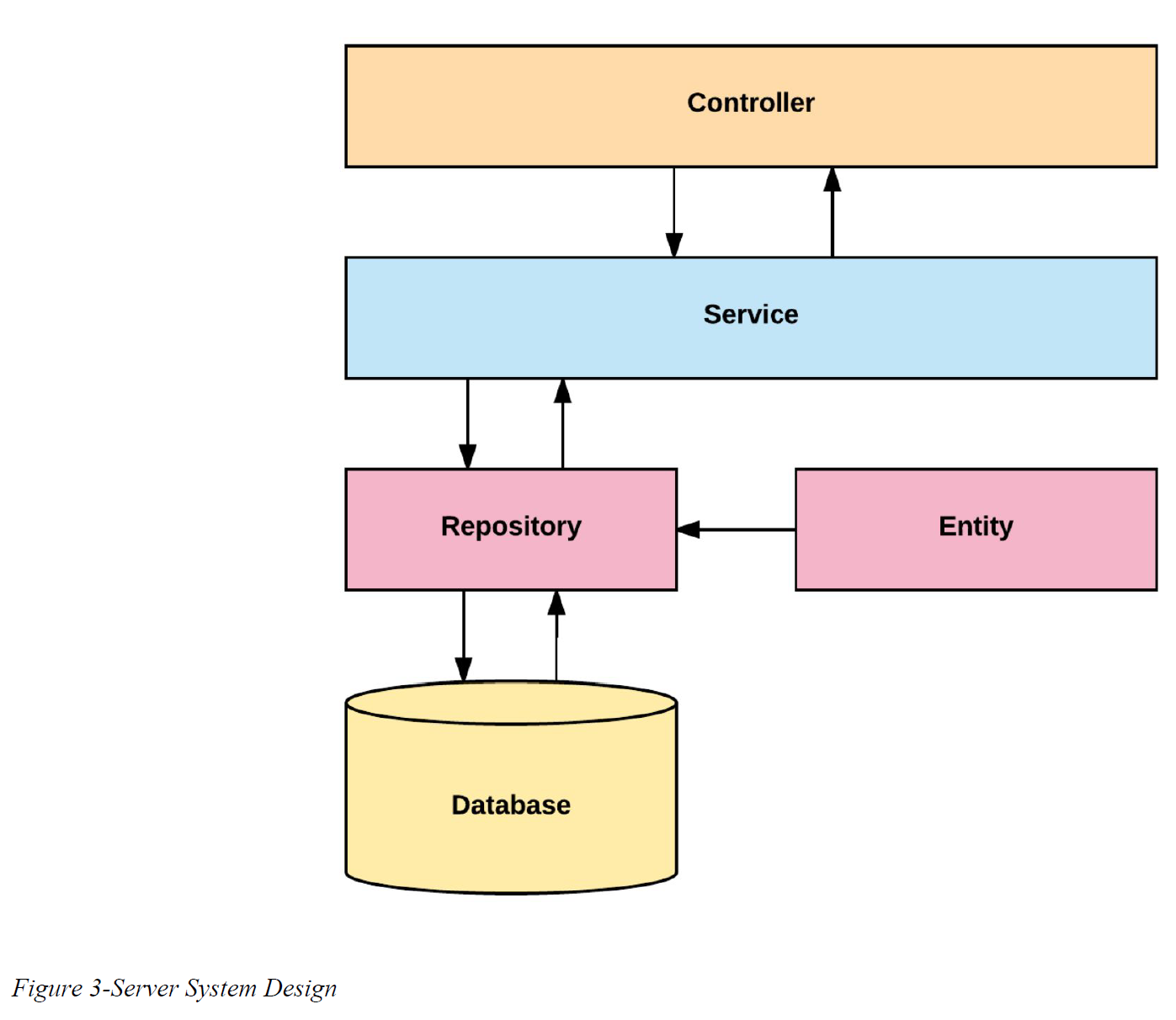
Client-Server is used in the system to deliver the application. A highly available system that can

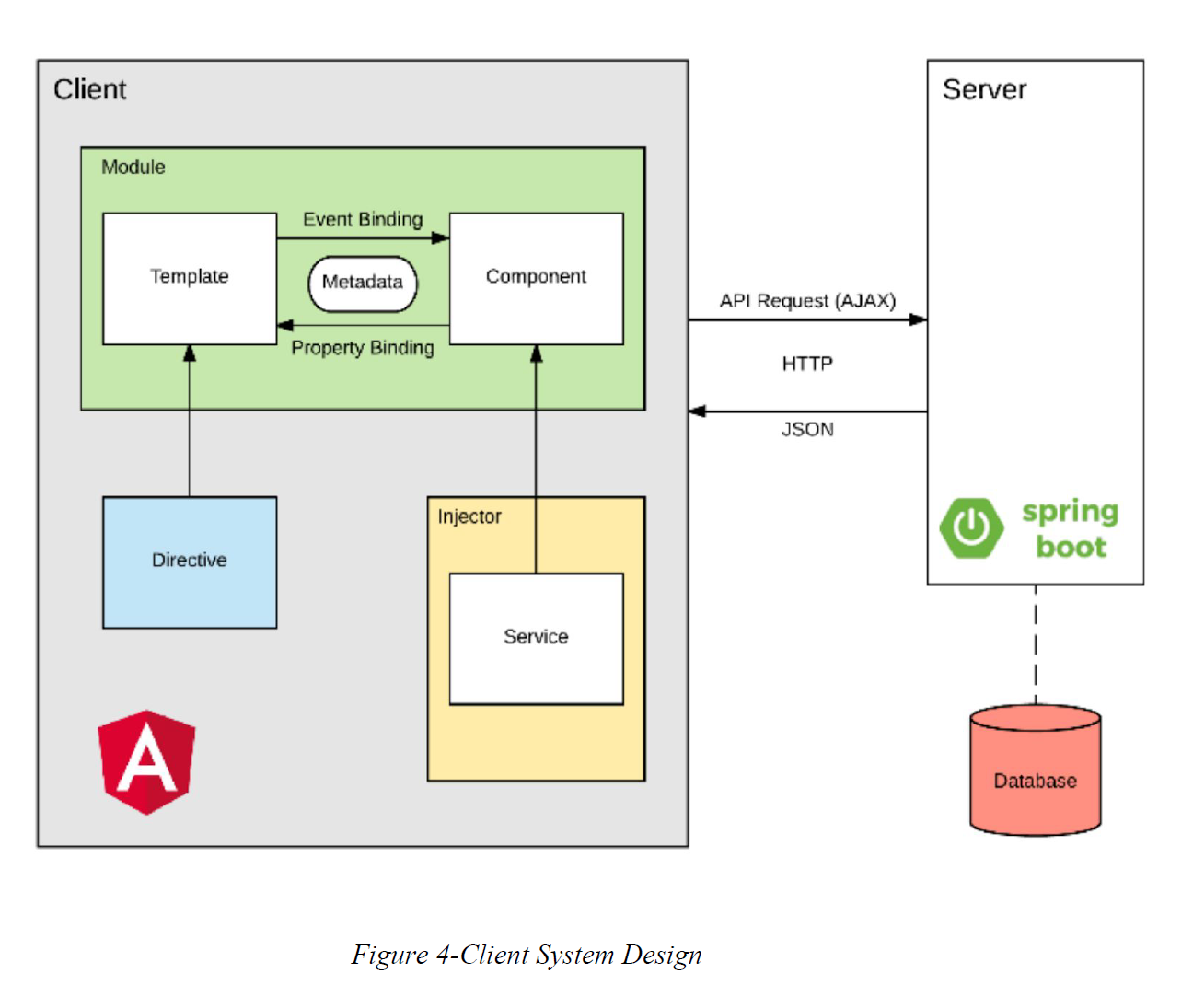
be consumed from several parties called for this design. It allows the centralization of the code

logic and database. We had to access the application from the web, iOS and Android devices so this was a very good choice.



Repository pattern is another design used in our system. It minimizes the amount of duplicate code in the system by abstracting the basic CRUD operations. It also ties the data entities to the domain model which favors development. The code would have to comply with the entity restrictions in order to even comply. It also helps maintain data integrity.





## System and Subsystem Decomposition

The system is made out of two major subsystems and two minor ones. They are broken

down as follows according to its tasks and interactions.

### Server Subsystem

* Interacts with the database.
* Serves all the requests from the web
* Handles server side security.
* Analyzes the statistics of the text.
* Performs OCR.
* Optimizes images.

### Client Subsystem

* Displays the application.
* Routes server-side API calls.
* Contains the main boundaries for user interaction.
* Client side data validation.

### Deployment Diagram

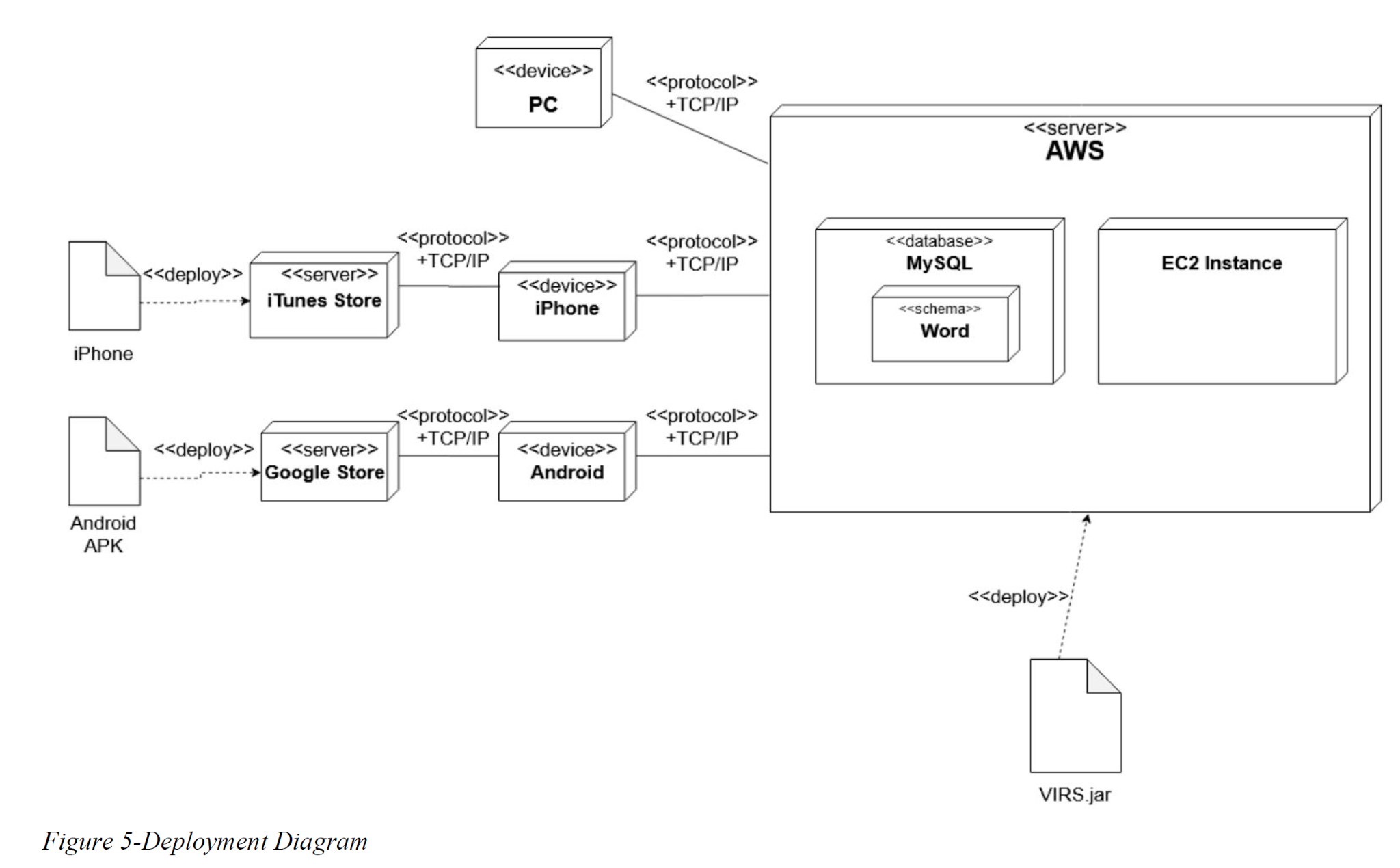
Deployment of the application consists in several steps run with a bash script. It starts in the

front end and it propagates to the backend and eventually to the final product. We are building

with ng for Angular products and Maven for the backend.

This is a detailed explanation for the development pipeline.

* Angular
  + Clean and build the application
  + Run karma tests
  + Run e2e
* Copy resources to the backend
* Spring
  + Increase version number
  + Clean and build
  + Run Unit tests
  + Run Integration tests
  + Create jar executable
* Bundle the application for AWS
* Upload to AWS servers.



## Design Patterns

The following design patterns were used in the application.

**Dependency Injection:** This is a core design pattern for Spring and Angular 4. It allows the

objects to be readily available and injected when needed rather than having to create them.

**Bridge:** This design pattern is used by the application to interact with libraries that are not

native. Ex: tesseract and OpenCV

**Front controller:** Controllers are one of the main Components in the MVC design.

**Marker:** All components in Spring are annotated with markers to further specify the correct

stereotype. This allows the initialization of the correct optimized components when the

application runs.

**Module:** The application is divided in several modules according to functionality. This

separation promotes organization and grouping of features.

# System Validation

**Test Case ID: TC01\_Frontend\_Dependencies**

Description: Frontend dependencies should be installed

Test steps:

1. Navigate to virs code/frontend directory

2. Run ‘npm install’ command

Expected result: Frontend installation success message

Actual result: As expected

Status(Pass/Fail): Pass

**Test Case ID: TC02\_Backend\_dependencies**

Description: Backend dependencies should be installed

Test steps:

1. Navigate to virs code directory

2. Run ‘virs.cmd install-only’ command

Expected result: Backend installation success message

Actual result: As expected

Status(Pass/Fail): Pass

**Test Case ID: TC05\_DOCX\_FILE\_UPLOAD**

Description: Test that the file upload service accepts .docx file uploads with no error.

Test steps:

1. Navigate to the myvirs.com dashboard

2. Select upload document

3. Select a .docx file for upload and hit enhanced text

4. Wait to see the screen change to the enhanced text display

Expected result: .docx file is uploaded with no errors, and the enhanced text screen is shown

Actual result: As expected

Status(Pass/Fail): Pass

**Test Case ID: TC06\_DOCX\_WORD\_OCR**

Description: Docx files that are uploaded have their text displayed correctly on the enhanced text screen

Test steps:

1. Navigate to the myvirs.com dashboard

2. Select upload document

3. Select a .docx file for upload and hit enhanced text

4. View the words; Scroll if necessary to see all the text from the uploaded document.

Expected result: Words from the chosen docx file appear on the screen

Actual result: As expected

Status(Pass/Fail): Pass

**Test Case ID: TC07\_PDF\_FILE\_UPLOAD**

Description: Test that the file upload service accepts .pdf file uploads with no error.

Test steps:

1. Navigate to the myvirs.com dashboard

2. Select upload pdf

3. Select a .pdf file to upload and hit enhanced text

4. Wait to see the screen change the enhanced text display

Expected result: PDF is uploaded with no errors, and the enhanced text screen is shown

Actual result: As expected

Status(Pass/Fail): Pass

**Test Case ID: TC08\_PDF\_WORD\_OCR**

Description: Test that the ENTIRE .pdf file uploads with no error.

Test steps:

1. Navigate to the myvirs.com dashboard

2. Select upload pdf

3. Select a .pdf file to upload and hit enhanced text

4. View the words; Scroll if necessary to see all the text from the uploaded .pdf file.

Expected result: The text from the chosen .pdf are displayed on the enhanced text screen

Actual result: As expected

Status(Pass/Fail): Pass

**Test Case ID: TC10\_NEW\_AWS\_DEPLOYMENT**

Description: Test that we can upload a new version of the code to AWS Elastic beanstalk and have the site updated.

Test steps:

1. Login to AWS Dashboard

2. Navigate to elastic beanstalk

3. Navigate to VocabularyInReading application

4. Navigate to application versions

5. Hit upload, fill out the information and select the file for the updated app

6. Wait for the site to upload and reload the application

7. Ensure that the site is running again and the changes made in the new version are reflected.

Expected result: New version of application is uploaded to AWS and the changes made in the update are visible on the site upon reload.

Actual result: As expected

Status(Pass/Fail): Pass

**Test Case ID: TC\_11\_Correct\_Mobile\_View**

Description: Verify that the mobile site no longer contains formatting errors.

Test steps:

1. Open up the website on a mobile device

2. Navigate to each section of the website and check that it looks the same as the desktop site.

3. Navigate to the bottom of the page and verify that no donation button is shown.

Expected result: The site will be correctly displayed on the desktop site and the mobile version of the site.

Actual result: As expected

Status(Pass/Fail): Pass

# Glossary

Note: Glossary didn’t change in VIRS v6.0 from previous versions

**Academic Word List (AWL):** List of words used in the natural English language with

frequency high enough but that does not make it to the high frequency list.

**High Frequency List:** List of words used in natural English language with a high frequency.

**Medium Frequency List:** List of words used in natural English language with a medium

frequency.

**Low Frequency List:** List of words used in natural English language with a low frequency.

**Flesch Reading Ease Score:** A test designed to calculate how hard a text is to understand in

English.

**Word Definition:** Meaning of a word as per Wiki Dictionary. It contains etymology, meaning,

and usage information.

**Category:** The assigned value to a word from one of the above lists. A word category can be:

AWL, High Frequency, Medium Frequency, and Low Frequency.

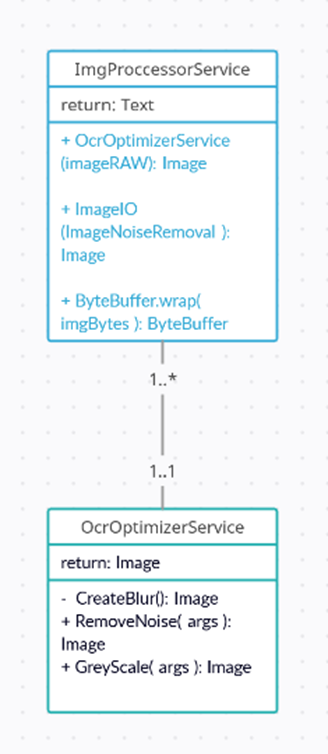
**Inflection:** A modification of a word to express additional meanings: plural and conjugations.

**School Dictionary:** A collection of all the lists. It can be used as a reference for the word

Categories.

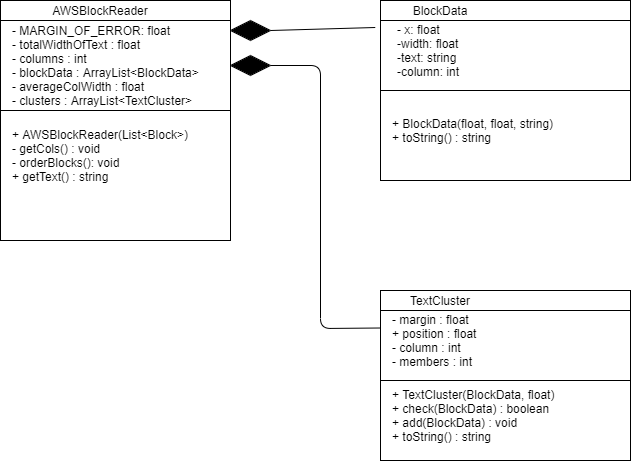
# Appendix

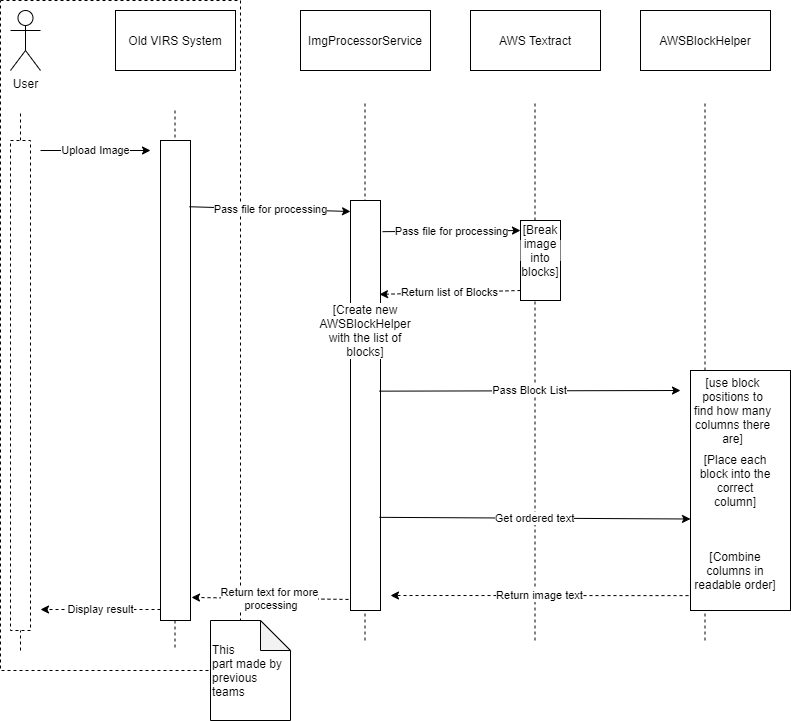
## Appendix A - UML Diagrams

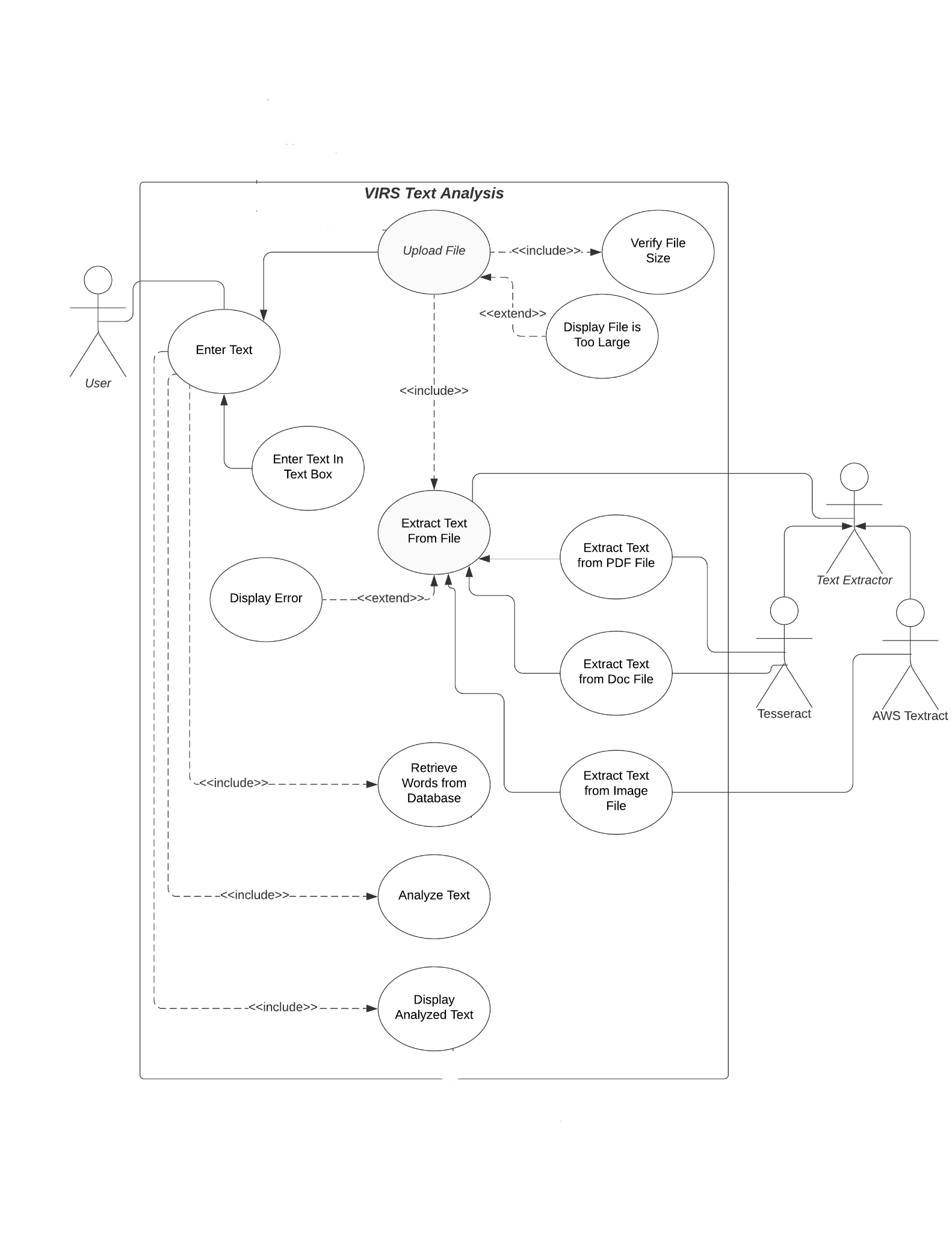


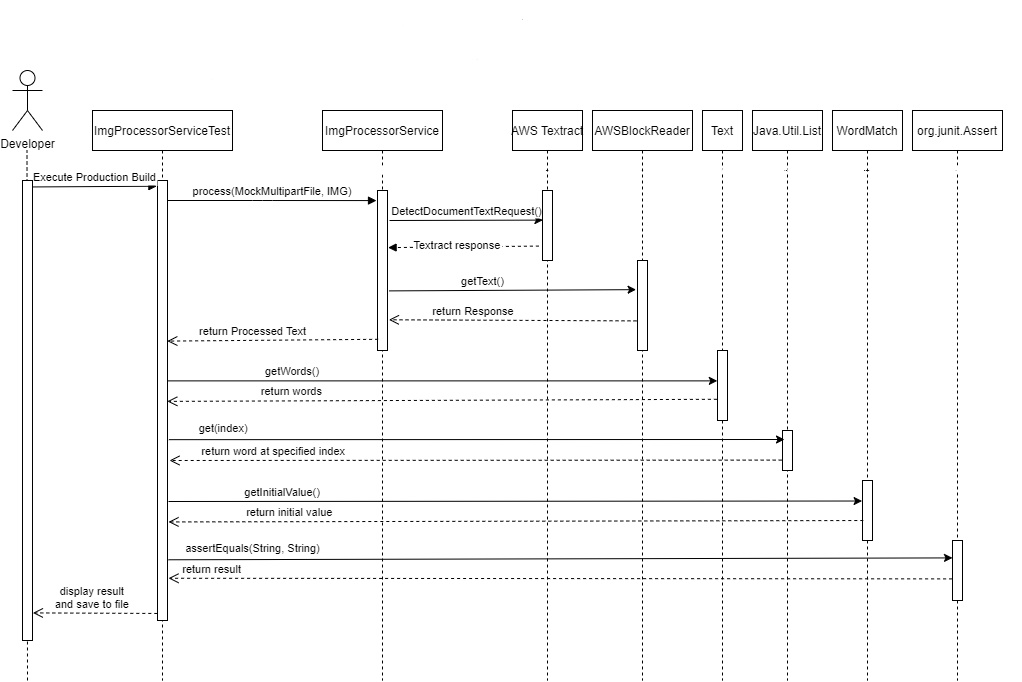
ImgProcessorService\_Diagram

Here are the systems that were created and revised in previous versions of VIRS.

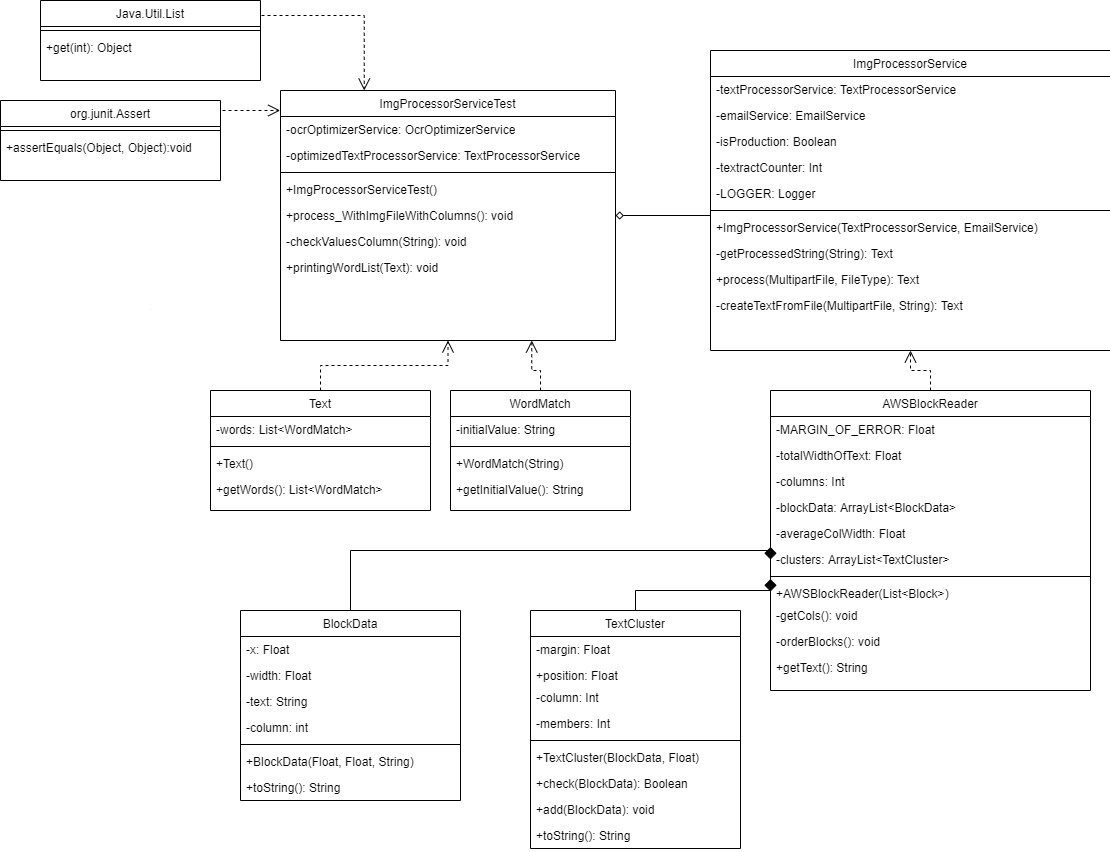






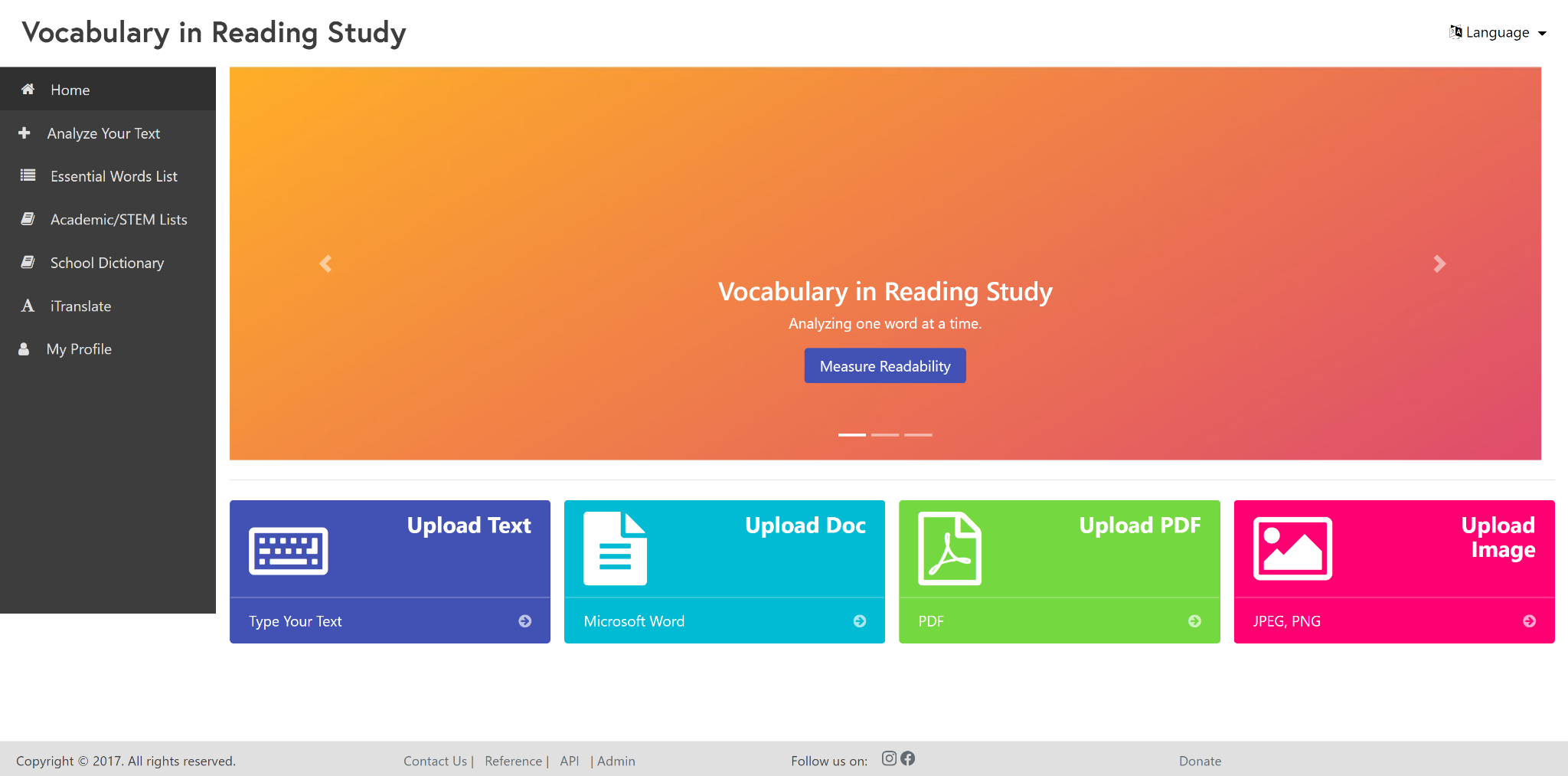


Major work was done in this area by Regine Donatien and Diego Chacin, Added an OCR Optimizer to service the IMG Processor Service in order to clean up image.

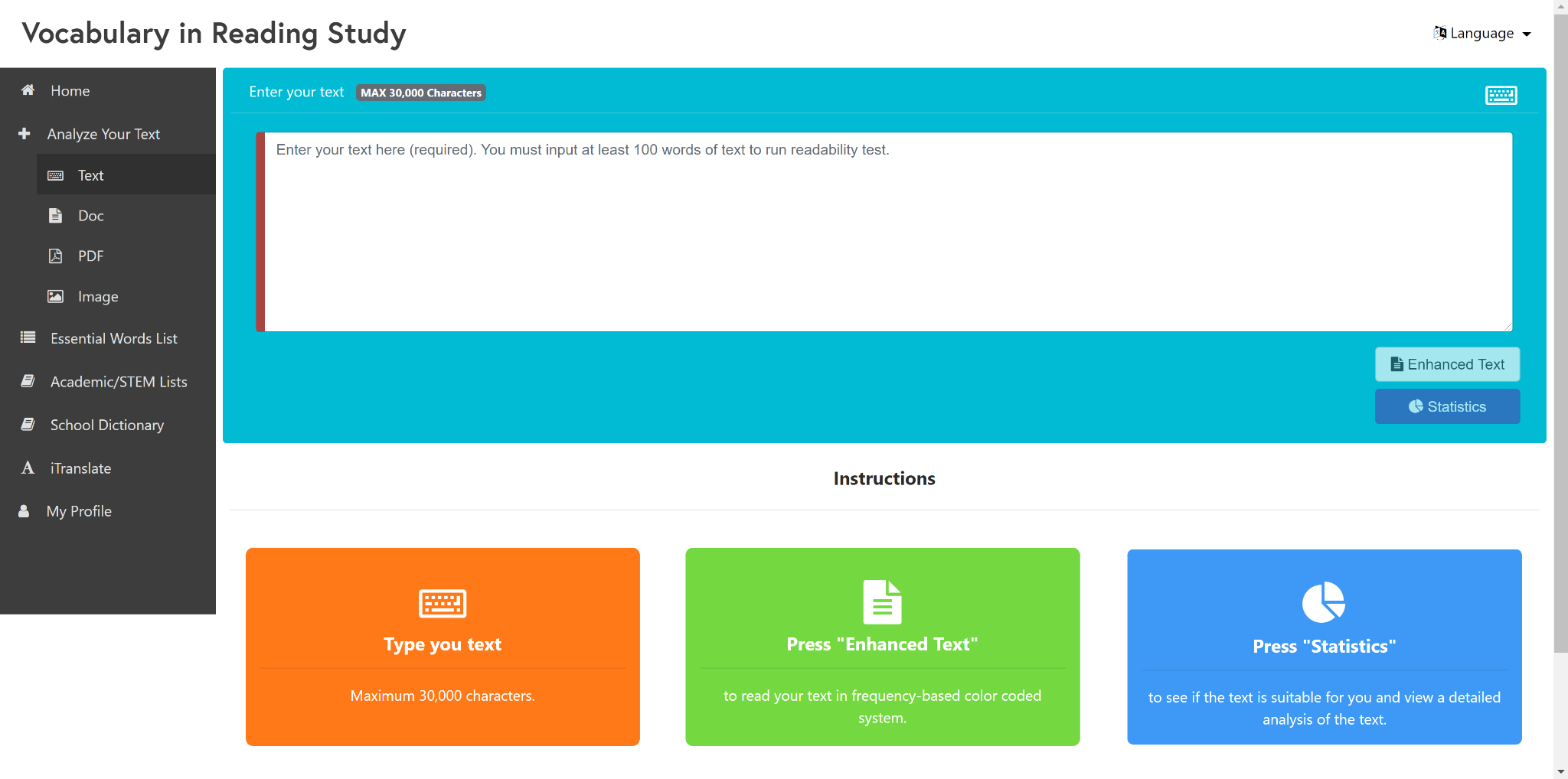


## Appendix B - User Interface Design

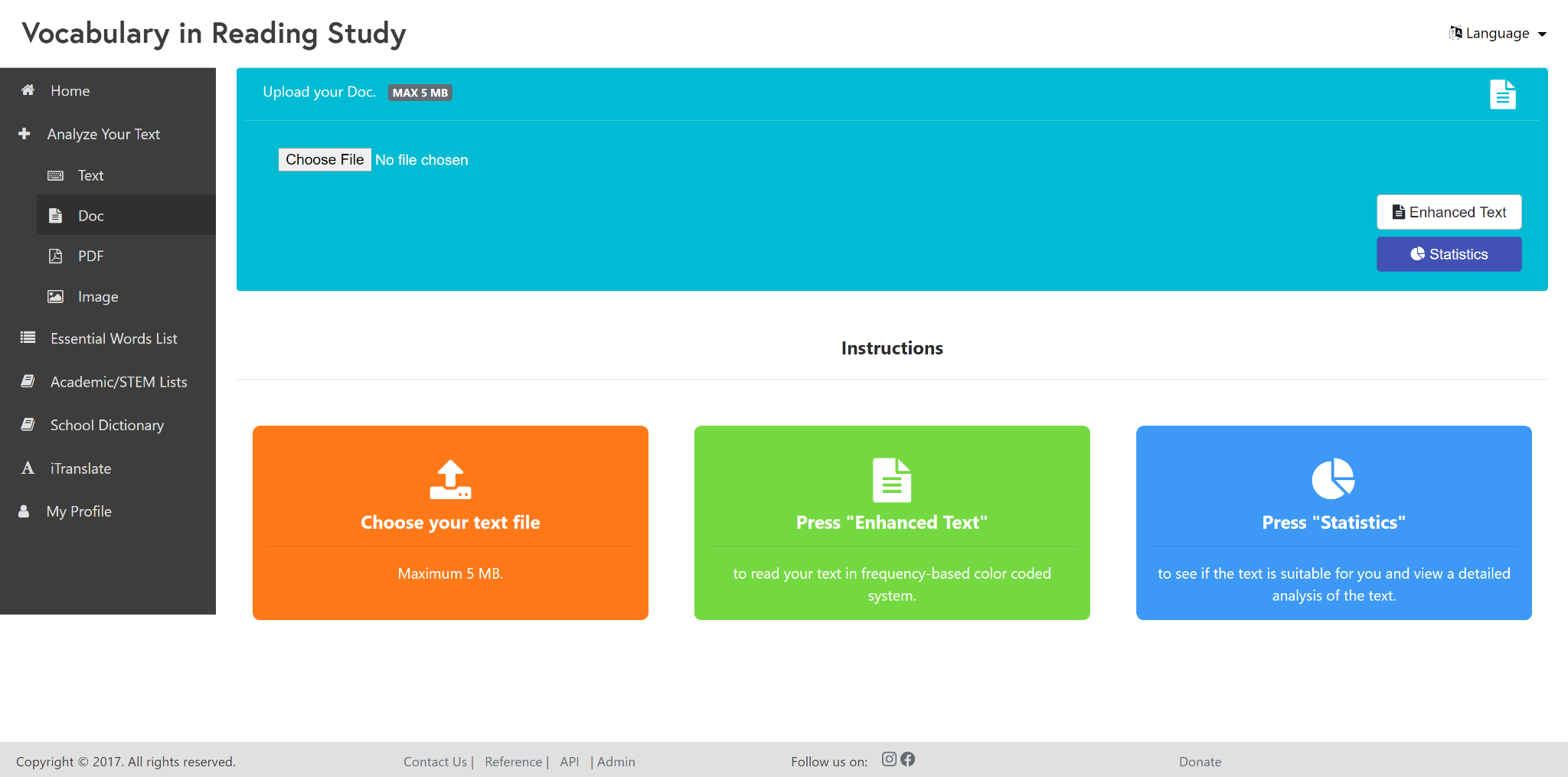
### Home Page



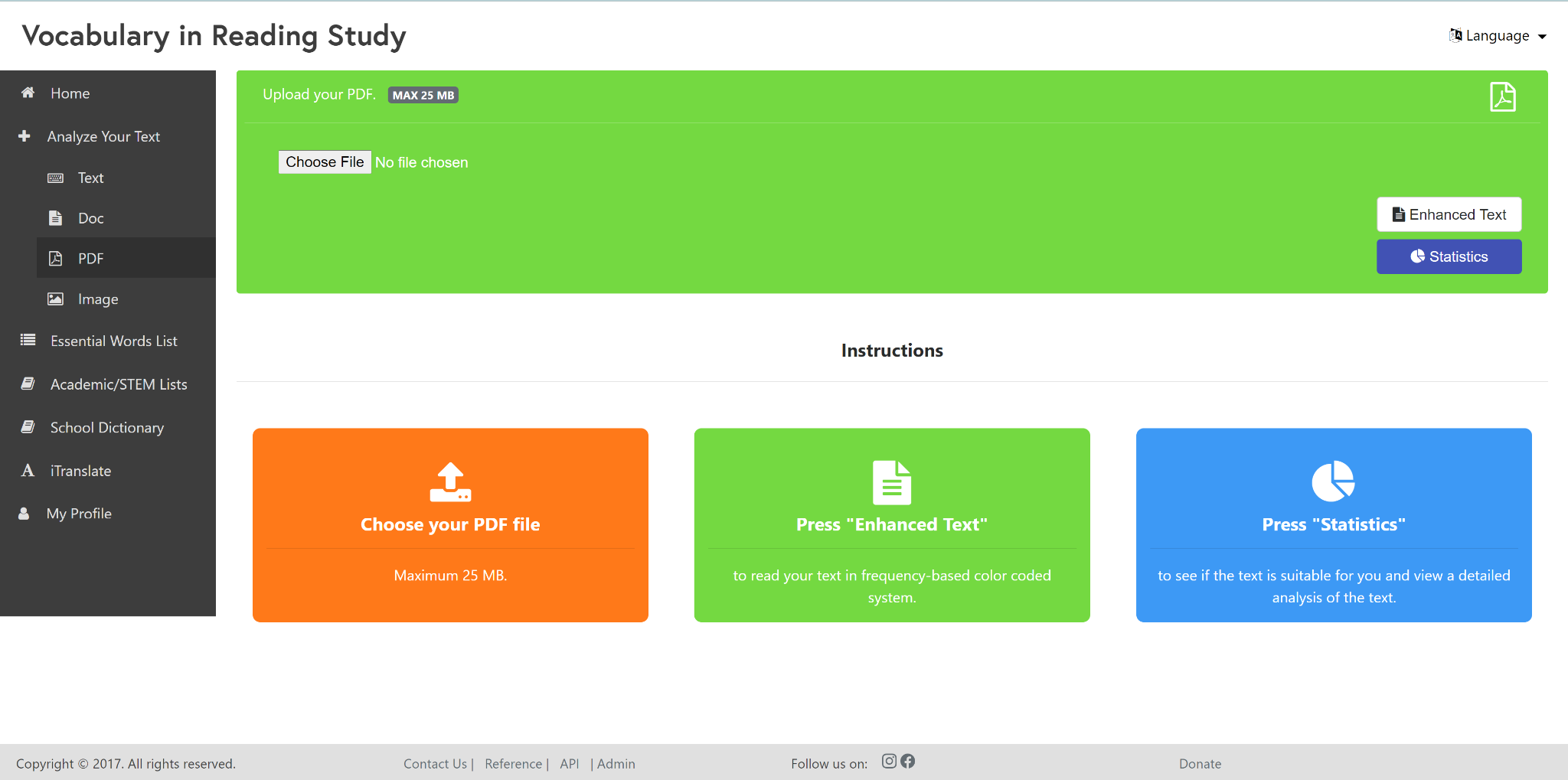
### Analyze Text



### Analyze Doc



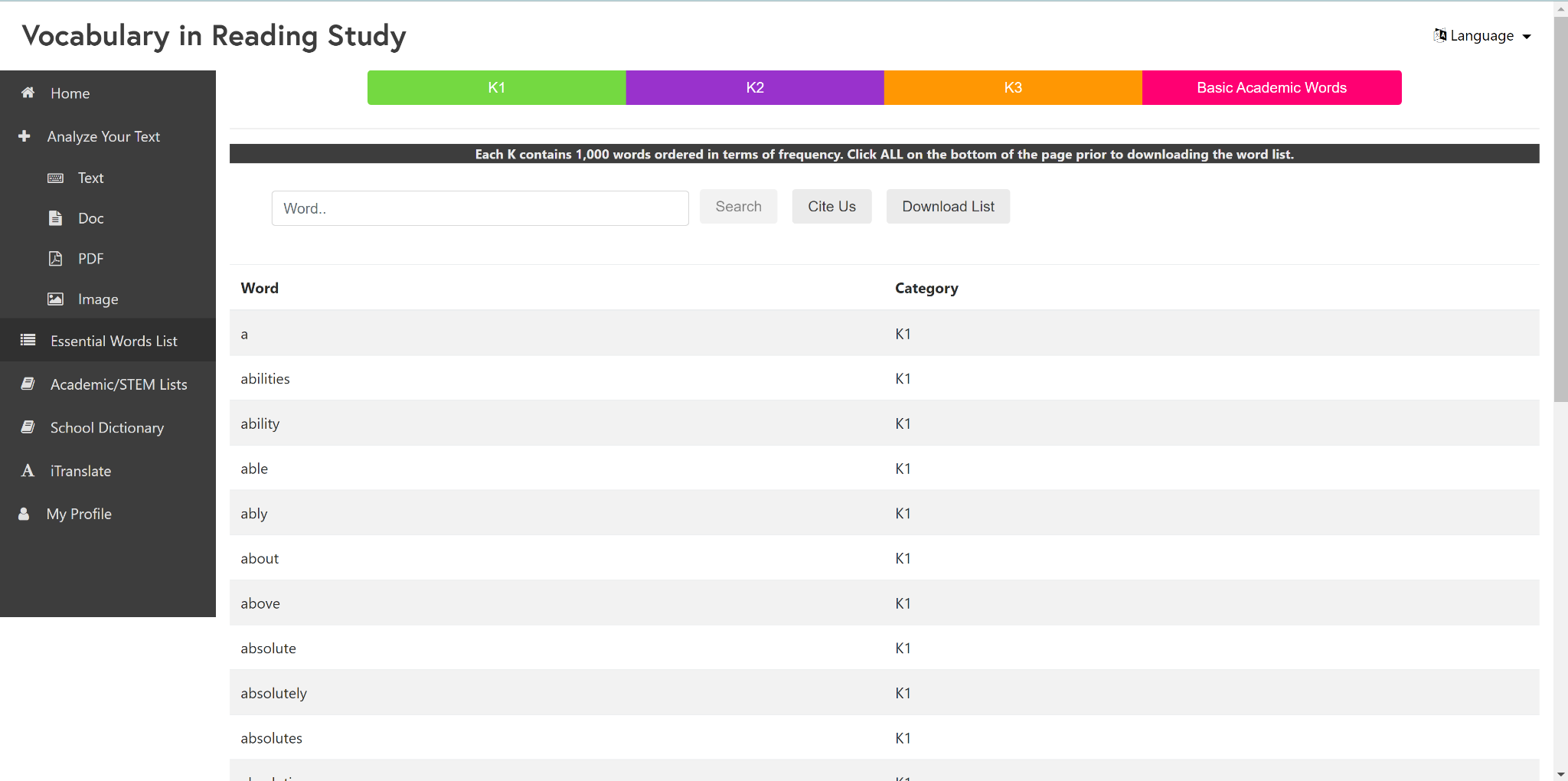
### Analyze PDF



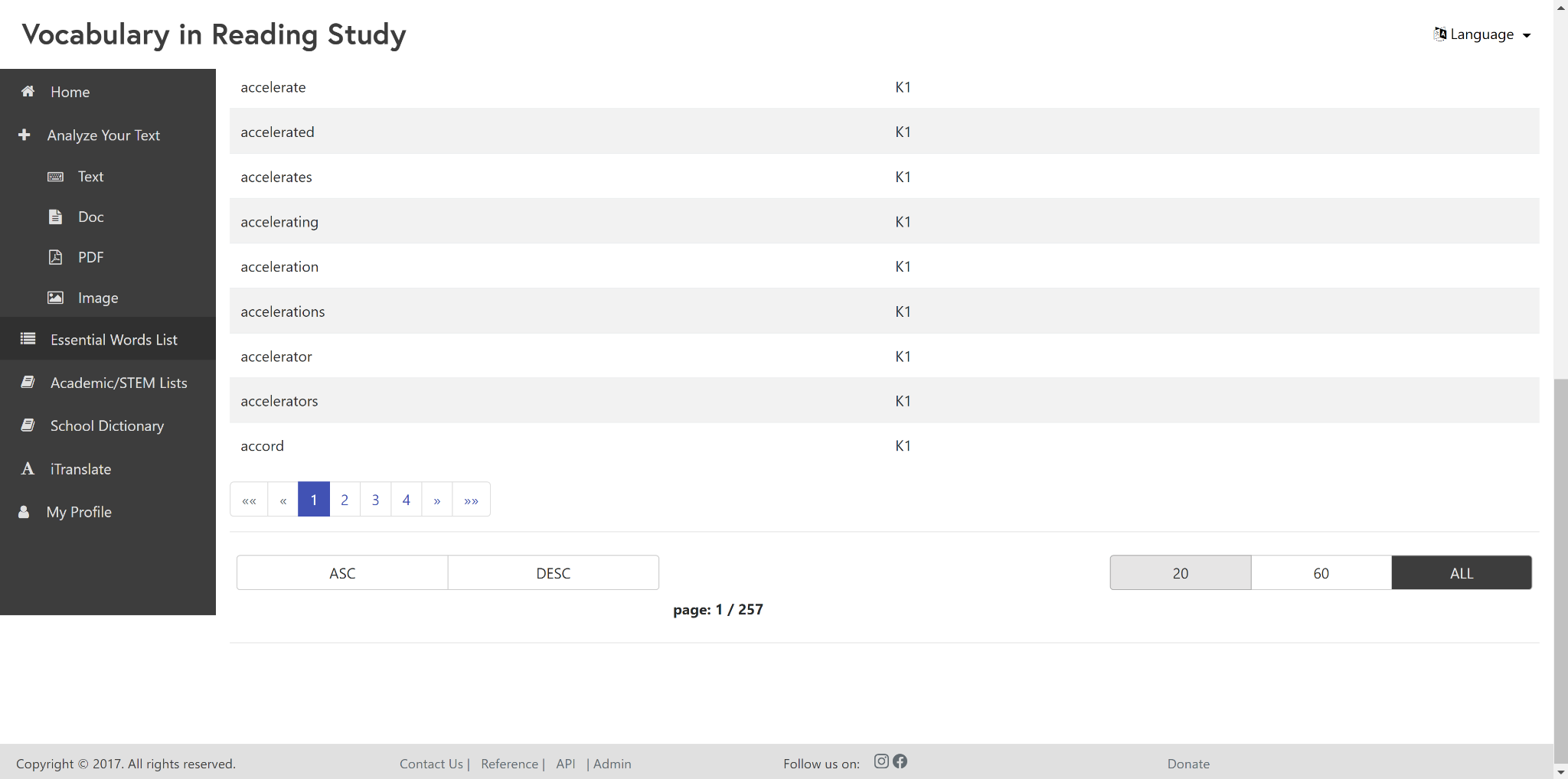
### Analyze Image



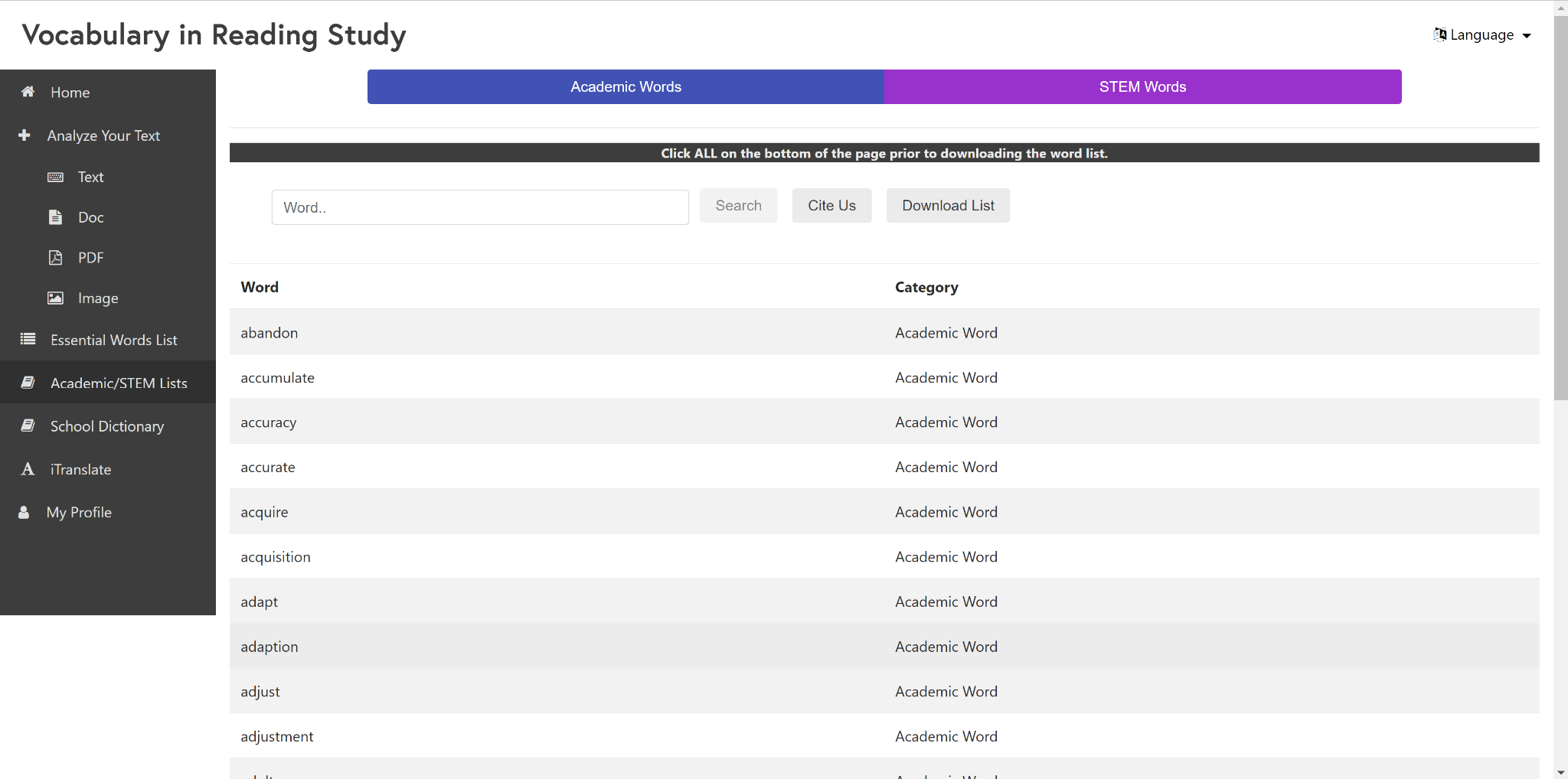
### Essential Word List - Top



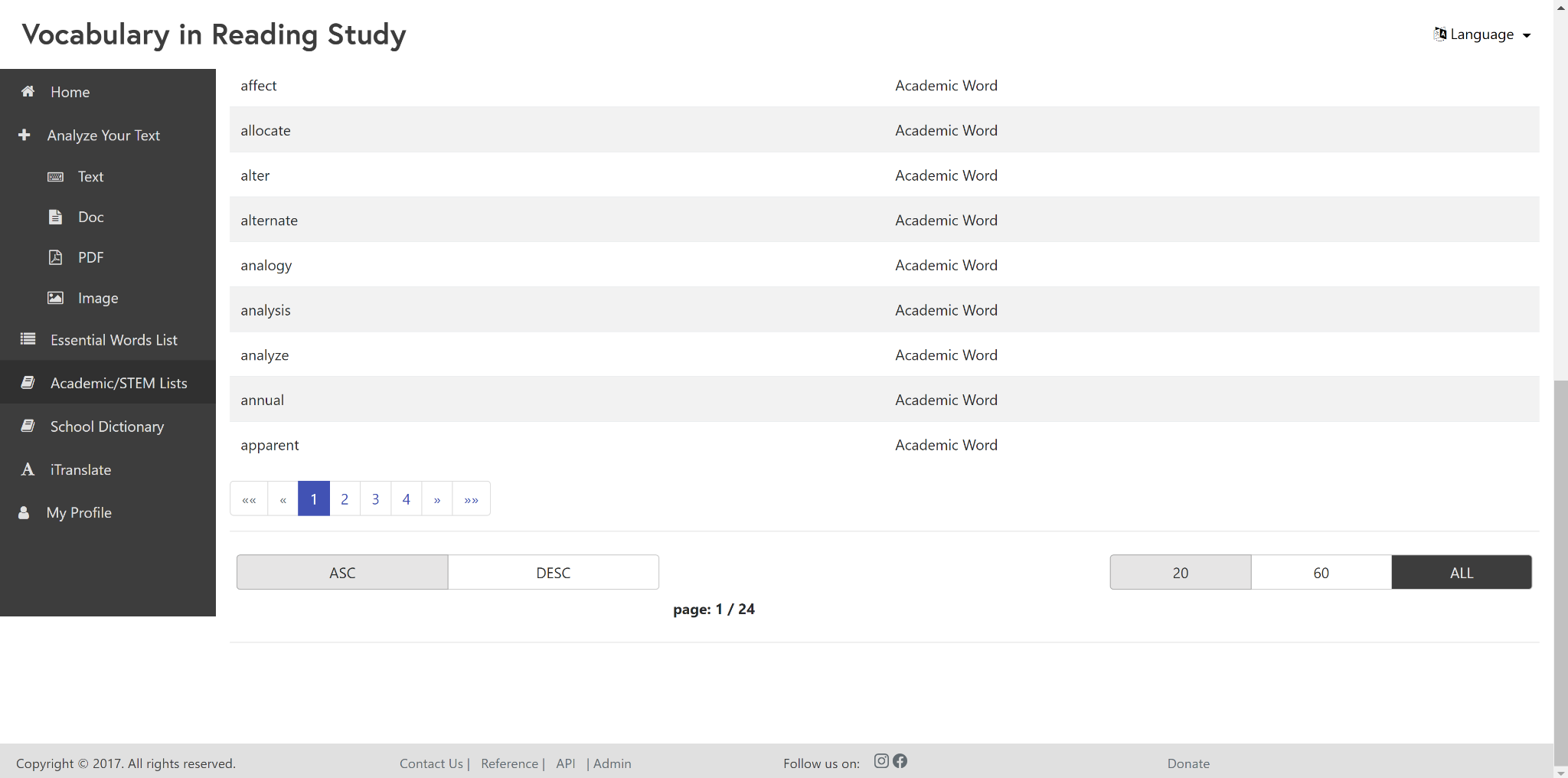
### Essential Word List - Bottom



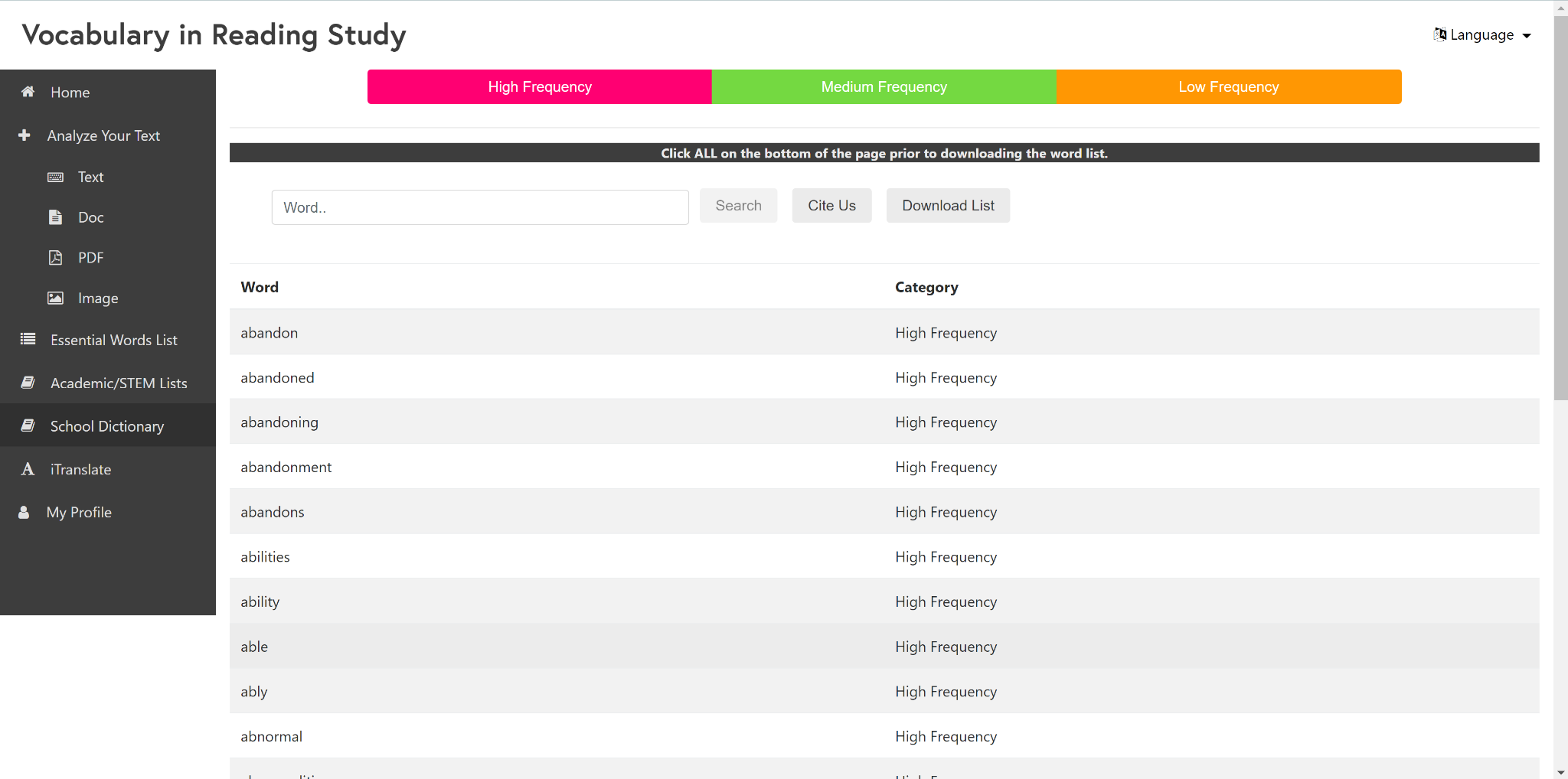
### Academic/STEM List - Top



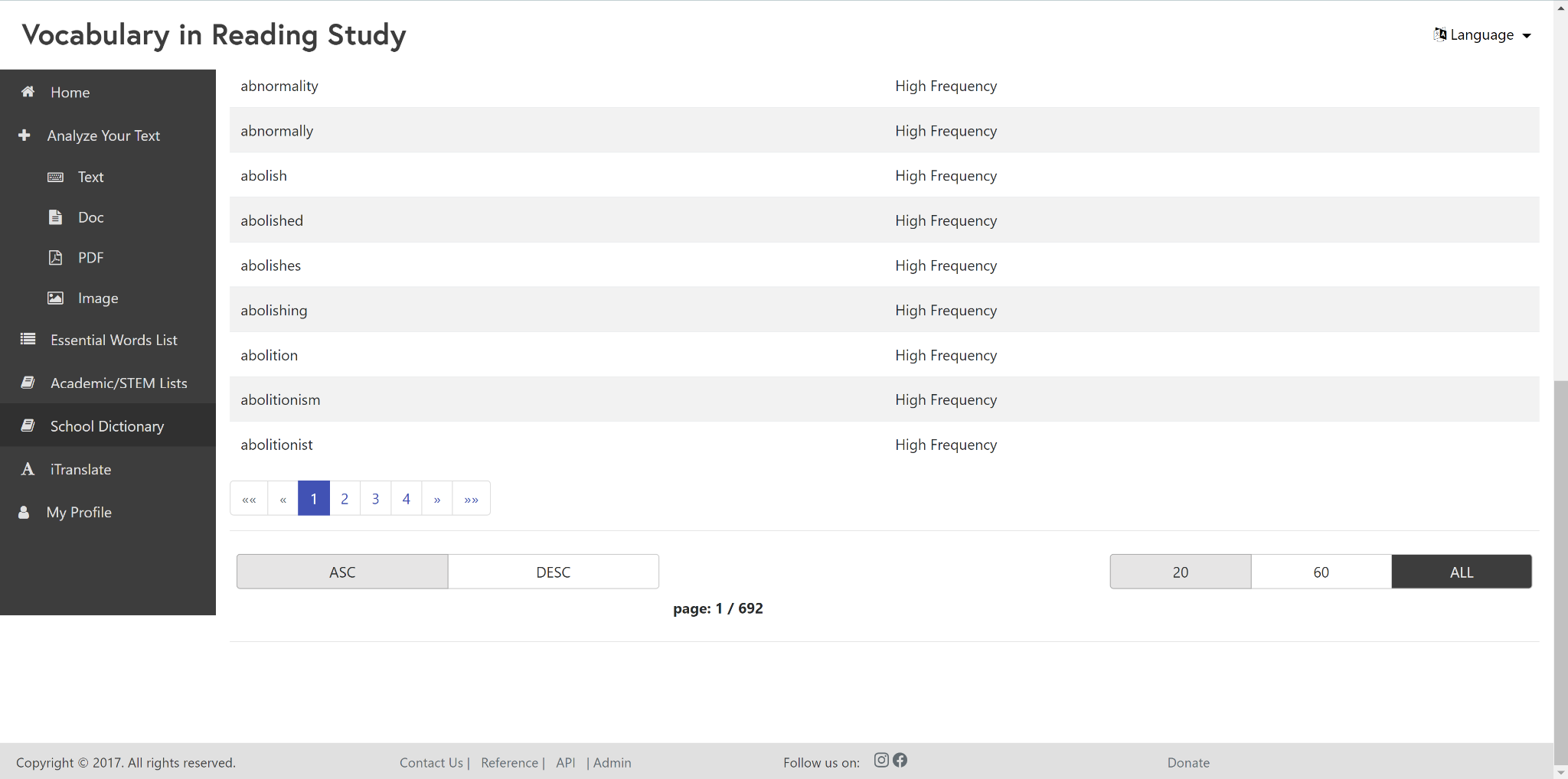
### Academic/STEM List - Bottom



### School Dictionary - Top



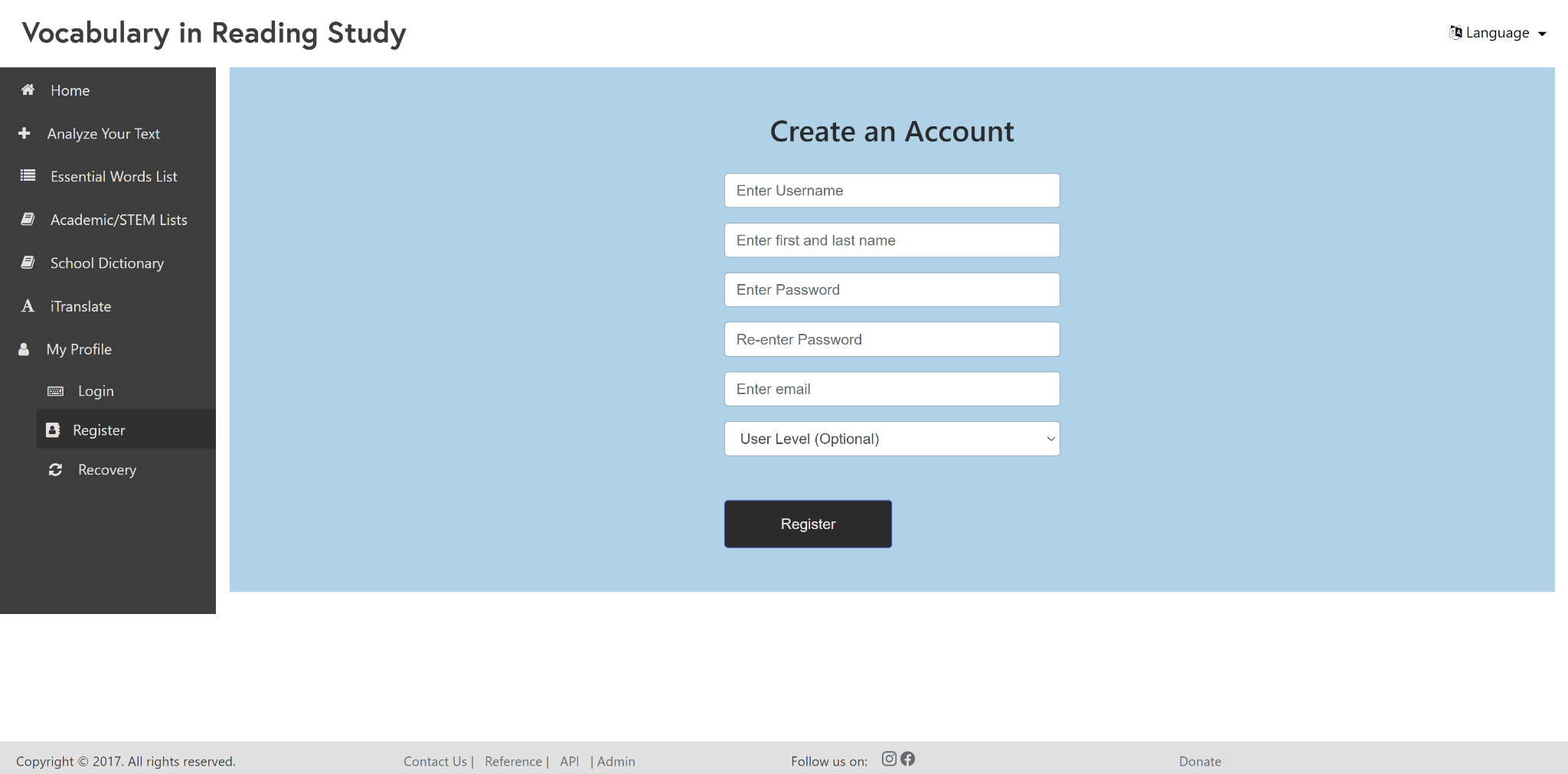
### School Dictionary - Bottom



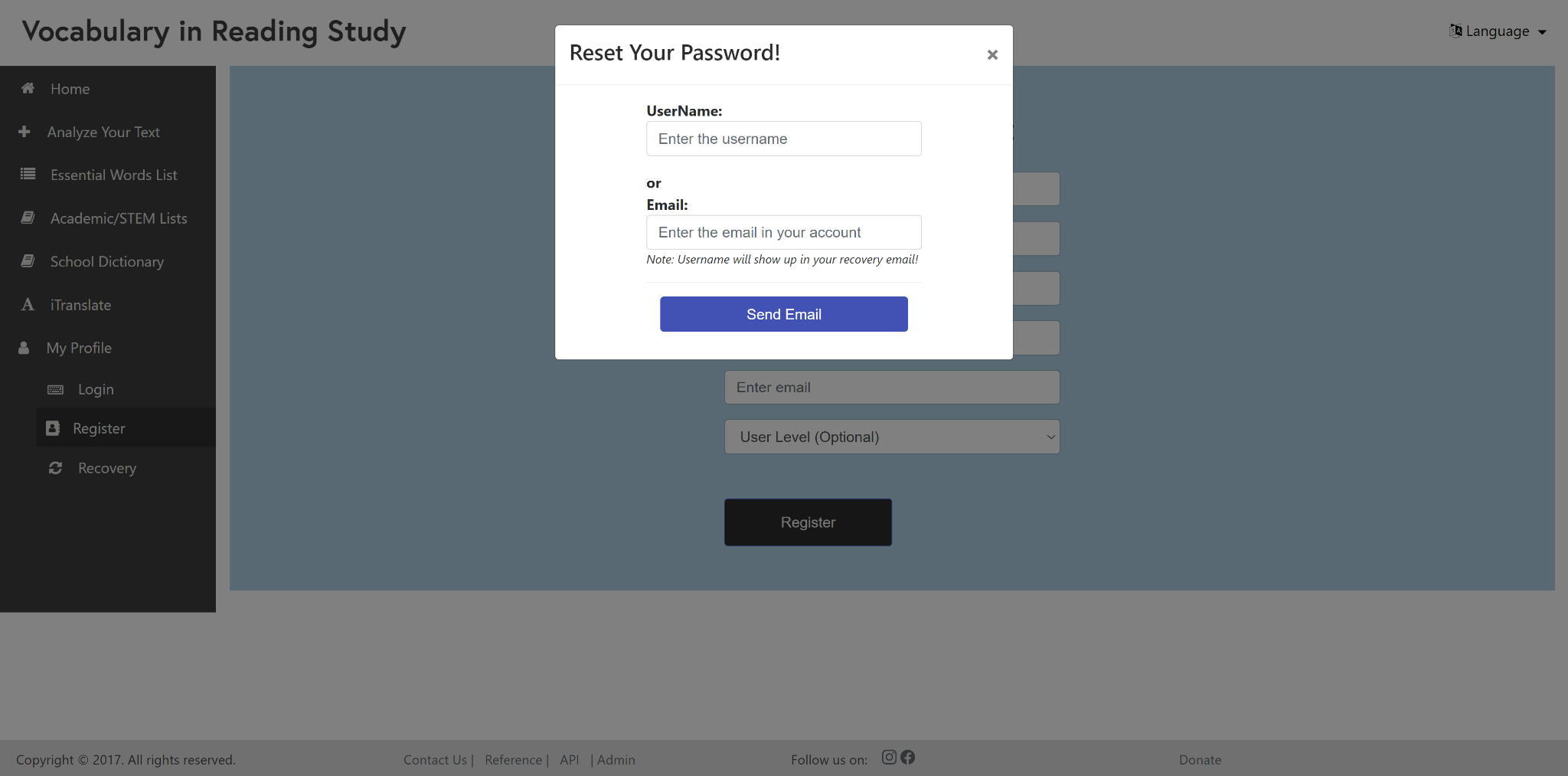
### iTranslate



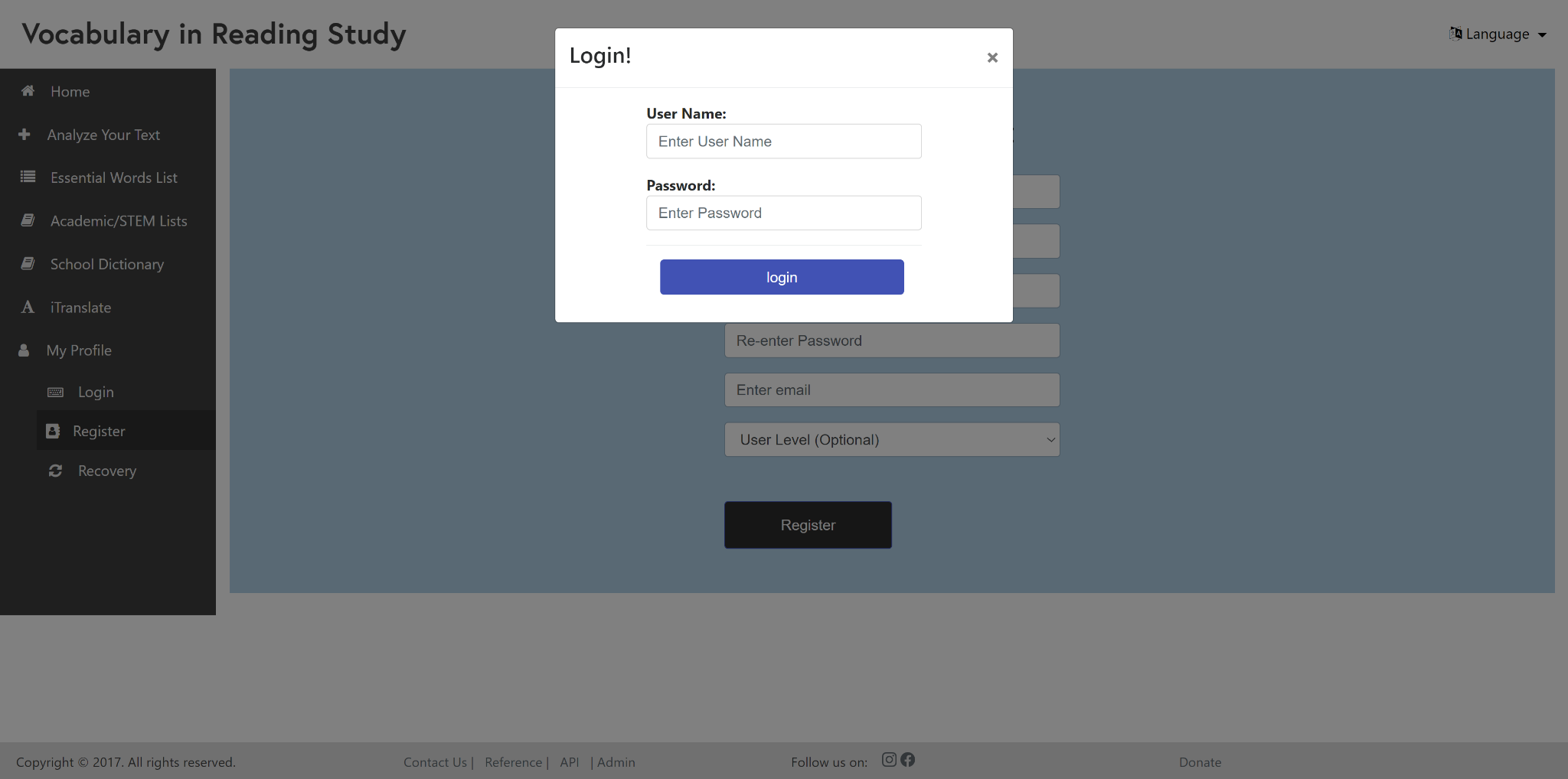
### Register



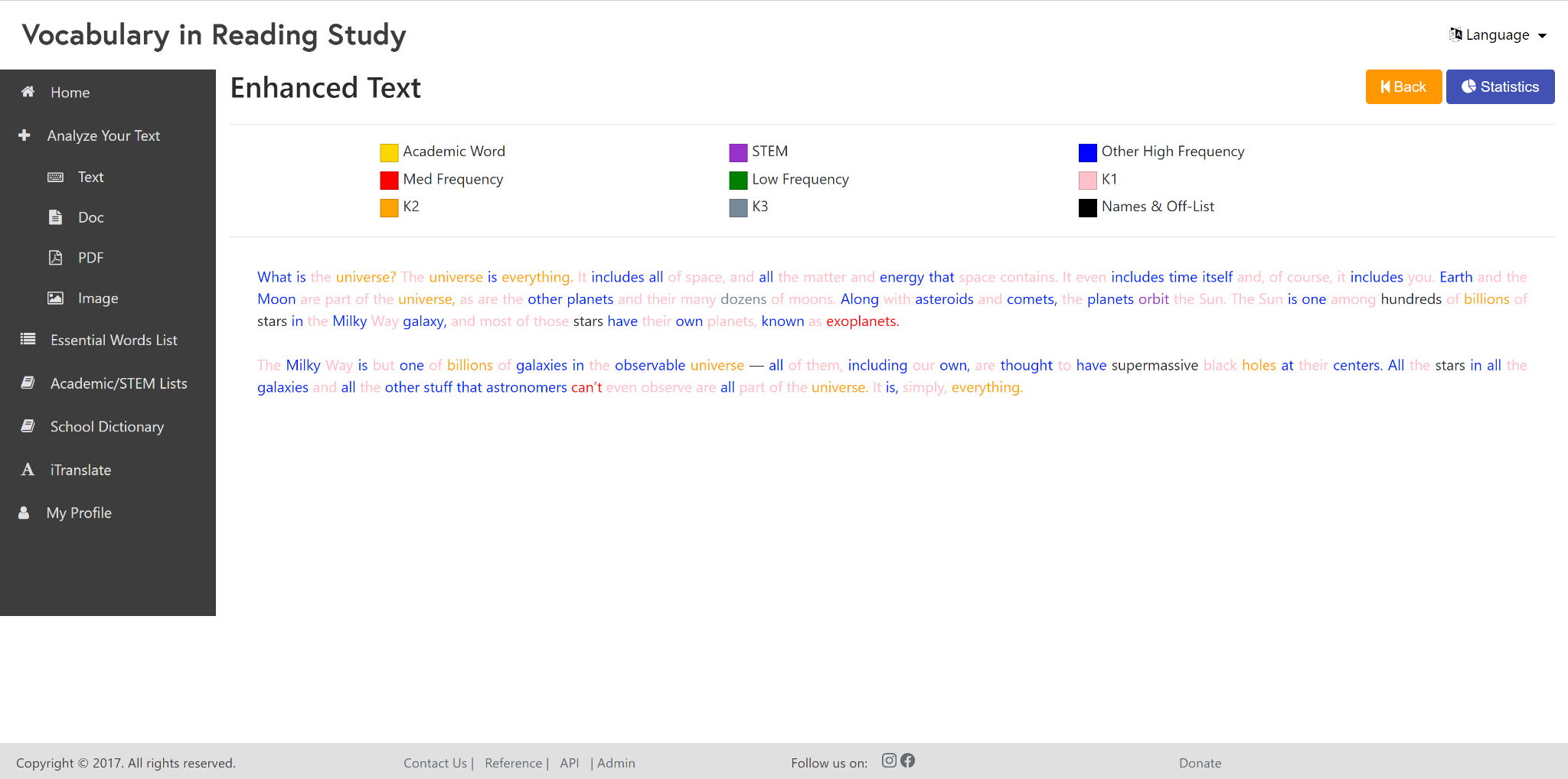
### Recovery



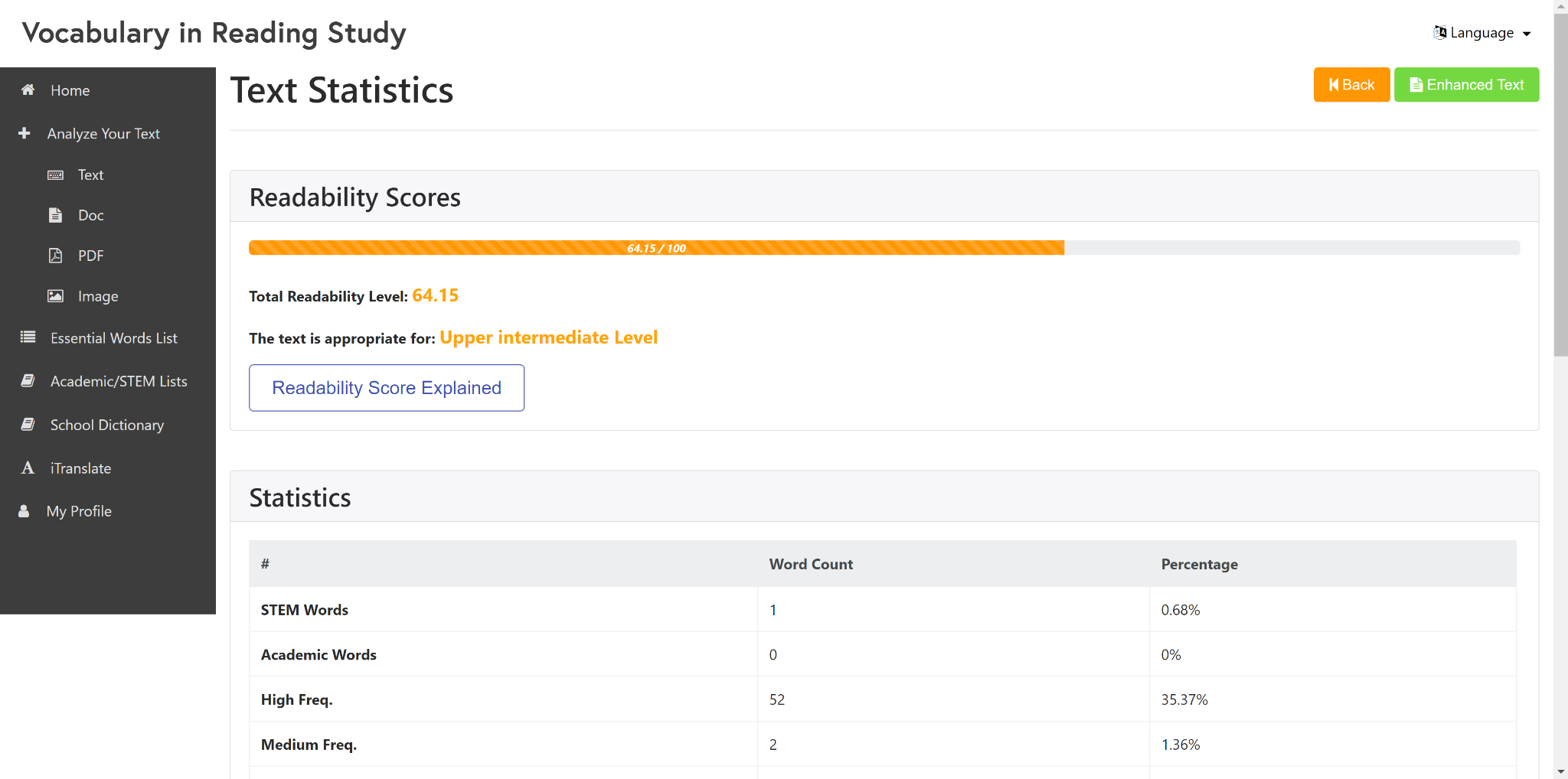
### Login



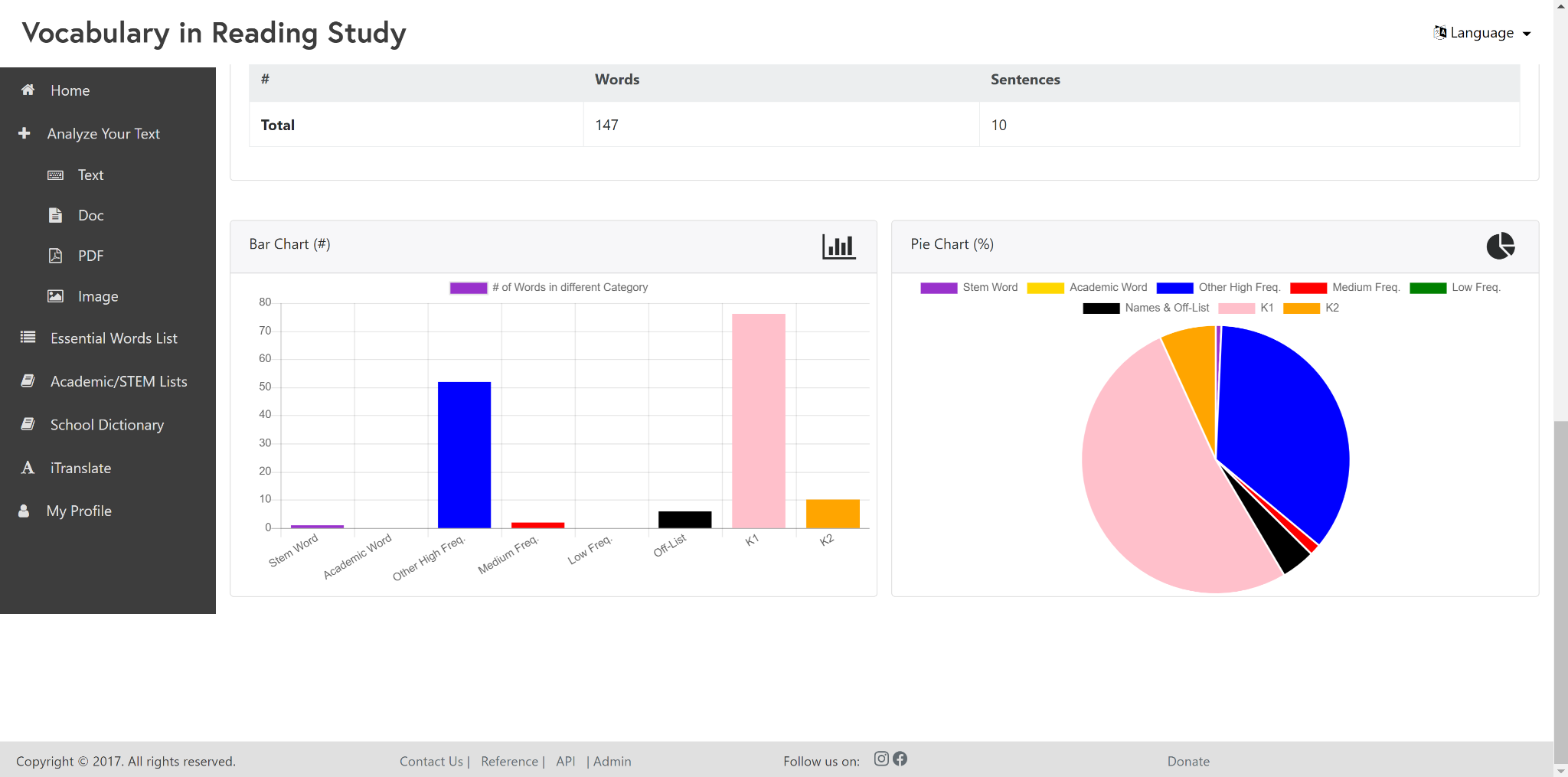
### Enhanced Text



### Statistics - Top



### Statistics - Bottom



# Appendix C - Sprint Review Reports

## Sprint 1

Sprint Review Meeting Minutes

Attendees: Diego Chacin

Regine Donatien

Javier Duenas

Jason Naya

Michelle Cruz

Start time: 9:00

End time: 9:30

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

Research on image recognition software to continue, possibly looking for a plug in rather than making new code.

Fixing the word lists to not include frequency

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting.

Improving the UX design of the website

## Sprint 2

Sprint Review Meeting Minutes

Attendees: Diego Chacin

Regine Donatien

Javier Duenas

Jason Naya

Michelle Cruz

Start time: 9:00

End time: 9:30

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

Research on image recognition software to continue, possibly looking for a plug in rather than making new code.

Fixing the word lists to not include frequency

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting.

Improving the UX design of the website

## Sprint 3

**Sprint Review Meeting Minutes**

Attendees: Diego Chacin

Regine Donatien

Javier Duenas

Jason Naya

Michelle Cruz

Start time: 9:00

End time: 9:30

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

* Research on image recognition software to continue, possibly looking for a plug in rather than making new code.
* Improving the UX design of the website

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting.

* N/A

## Sprint 4

N/A

## Sprint 5

**Sprint Review Meeting Minutes**

Attendees: Diego Chacin

Regine Donatien

Javier Duenas

Jason Naya

Michelle Cruz

Start time: 9:00

End time: 9:30

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

* Research on image recognition software to continue, possibly looking for a plug in rather than making new code.
* Improving the UX design of the website
* Downloading the word lists
* Looking further into the workings of tesseract and textrac, add code to blue image

The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting.

* N/A

## Sprint 6

**Sprint Review Meeting Minutes**

Attendees: Diego Chacin

Regine Donatien

Javier Duenas

Jason Naya

Michelle Cruz

Start time: 9:00

End time: 9:30

After a show and tell presentation, the implementation of the following user stories were accepted by the product owners: All.

* Improving column recognition
* The following ones were rejected and moved back to the product backlog to be assigned to a future sprint at a future Sprint Planning meeting.
* N/A