

Test Plan

Overall Test Plan

Our testing approach will be to utilize both manual and automatic testing using the Vue frontend testing library and mocha and chai. We plan to test both successful and unsuccessful cases to ensure the app works as intended and error handling is also correct to ensure the best user experience. We will also be testing our APIs to ensure that they work correctly. We plan to use automated testing so that the tests can all be run with each push to our repository to ensure that the app continues to work and we will catch any unintended errors caused by changes in adding new features.

Test Case Descriptions

UC1.1 **User Creation Test 1**

UC1.2 Ensure user can create account using site UI.

UC1.3 Using Vue, fill in username, password, and confirm password with valid information. Submit form. Confirm account creation.

UC1.4 Input: Username, Password, Confirmed Password (identical to Password)

UC1.5 Output: Successful account creation - confirm by ensuring navigation to new account page

UC1.6 Normal

UC1.7 Blackbox

UC1.8 Functional

UC1.9 Integration

UC2.1 **User Creation Test 2**

UC2.2 Ensure proper behavior of User Creation API

UC2.3 Using Mocha Chai send request to user creation API with valid information, and confirm successful user creation.

UC2.4 Input: Username, Password

UC2.5 Output: Successful account creation return message

- UC2.6 Normal
- UC2.7 Blackbox
- UC2.8 Functional
- UC2.9 Unit

UC3.1 **User Creation Test 3**

- UC3.2 Ensure graceful failure and error handling when given invalid information for User Creation
- UC3.3 Using Vue, fill in username containing invalid characters, password, and confirm password. Submit form. Confirm inability to submit.
- UC3.4 Input: Username containing invalid characters, Password, Confirmed Password
- UC3.5 Output: error message displayed on sign up page stating invalid characters
- UC3.6 Abnormal
- UC3.7 Blackbox
- UC3.8 Functional
- UC3.9 Unit

UC4.1 **User Creation Test 4**

- UC4.2 Ensure graceful failure and error handling when given invalid information for User Creation
- UC4.3 Using Vue, fill in username, password, and confirm password that is different from original password. Submit form. Confirm inability to submit.
- UC4.4 Input: Username, Password, Confirmed Password that is different from original password
- UC4.5 Output: error message displayed on sign up page stating mismatched passwords
- UC4.6 Abnormal
- UC4.7 Blackbox
- UC4.8 Functional
- UC4.9 Unit

UL1.1 **User Login Test 1**

- UL1.2 Ensure user can log in with previously created account
- UL1.3 Using Vue, fill in username and password of account in database. Submit form.
- UL1.4 Input: Username, Password
- UL1.5 Output: Successful account authentication - confirm by ensuring navigation to

home page

UL1.6 Normal

UL1.7 Blackbox

UL1.8 Functional

UL1.9 Integration

UL2.1 User Login Test 2

UL2.2 Ensure proper behavior of User Login API

UL2.3 Using Mocha Chai send request to user login API with valid information, and confirm successful user authentication.

UL2.4 Input: Username, Password

UL2.5 Output: Successful account authentication returns message

UL2.6 Normal

UL2.7 Blackbox

UL2.8 Functional

UL2.9 Unit

UL3.1 User Login Test 3

UL3.2 Ensure graceful failure and error handling when given invalid information for User login

UL3.3 Using Vue, fill in username containing invalid characters and correct password. Submit form. Confirm inability to submit.

UL3.4 Input: Username containing invalid characters and correct Password

UL3.5 Output: error message displayed on sign up page stating invalid characters

UL3.6 Abnormal

UL3.7 Blackbox

UL3.8 Functional

UL3.9 Unit

SC1.1 Set Creation Test 1

SC1.2 Ensure user can create word sets

SC1.3 Using Vue, select words from dictionary and submit

SC1.4 Input: selection of words and submit button

SC1.5 Output: successful set creation page

SC1.6 Normal

SC1.7 Blackbox

SC1.8 Functional
SC1.9 Integration

SC2.1 Set Creation Test 2

SC2.2 Ensure user can create a new word in set creation page
SC2.3 Using Vue, enter word in word creation field on set creation page. Submit. Verify word creation and inclusion in dictionary list.
SC2.4 Input: word input and select submit button
SC2.5 Output: successful word creation and inclusion in dictionary list
SC2.6 Normal
SC2.7 Blackbox
SC2.8 Functional
SC2.9 Unit

SC3.1 Set Creation Test 3

SC3.2 Ensure user cannot create empty sets
SC3.3 Using Vue, submit new set without selecting any words from dictionary
SC3.4 Input: selection of submit button
SC3.5 Output: error message directing user to select words before submitting
SC3.6 Abnormal
SC3.7 Blackbox
SC3.8 Functional
SC3.9 Integration

WA1.1 Word Addition

WA1.2 Ensure that a word can be successfully created by a user
WA1.3 Using Vue, enter word in a specified, verify that the information provided by the dictionary api is correct, select any sets to add the word to, submit
WA1.4 Input: Word in specified language, sets to add it to.
WA1.5 Output: Successful creation of word
WA1.6 Normal
WA1.7 Blackbox
WA1.8 Functional
WA1.9 Integration

FC1.1 Flashcards Test 1

- FC1.2 Ensure Flash cards flip
- FC1.3 In flash cards page select the flash card, ensure the card changes to displaying the proper description information of the flash card
- FC1.4 Input: selection of the flash card
- FC1.5 Output: flash card flips and displays word description
- FC1.6 Normal
- FC1.7 Blackbox
- FC1.8 Functional
- FC1.9 Unit

FC1.1 **Flashcards Test 2**

- FC1.2 Ensure flash cards change when change card arrow is selected
- FC1.3 In flash cards page select the arrow to change flash card, ensure the card changes to the next flash card
- FC1.4 Input: selection of the change flash card button
- FC1.5 Output: displaying of next flash card
- FC1.6 Normal
- FC1.7 Blackbox
- FC1.8 Functional
- FC1.9 Unit

UD1.1 **User Dictionary 1**

- UD1.2 Ensure user can edit words in their dictionary
- UD1.3 Using Vue, select a word to edit, edit any desired information, submit.
- UD1.4 Input: Word to edit
- UD1.5 Output: Successfully edited word
- UD1.6 Normal
- UD1.7 Blackbox
- UD1.8 Functional
- UD1.9 Unit

UD1.1 **User Dictionary 2**

- UD1.2 Ensure user can delete words from their dictionary
- UD1.3 Using Vue, select a word to delete, click the delete button, confirm deletion.
- UD1.4 Input: Word to edit
- UD1.5 Output: Successfully edited word

UD1.6 Normal
UD1.7 Blackbox
UD1.8 Functional
UD1.9 Unit

PP1.1 Pronunciation Practice Test 1

PP1.2 Ensure successful return from pronunciation API with valid input
PP1.3 Pass audio file, which is known to be valid, to pronunciation API. Confirm successful return.
PP1.4 Input: Valid audio file
PP1.5 Output: successful return
PP1.6 Normal
PP1.7 Blackbox
PP1.8 Functional
PP1.9 Unit

PP2.1 Pronunciation Practice Test 2

PP2.2 Ensure error return from pronunciation API with invalid input
PP2.3 Pass audio file, which is known to be invalid, to pronunciation API. Confirm error return.
PP2.4 Input: Invalid audio file
PP2.5 Output: erroneous return
PP2.6 Abnormal
PP2.7 Blackbox
PP2.8 Functional
PP2.9 Unit

VG1.1 Vocab Game Test 1

VG1.2 Ensure vocab game ends when 3 translations are entered incorrectly during the game
VG1.3 While playing a game with a set, the game should end when lives are depleted. Incorrectly translate 3 words and the game ends, displaying the end score.
VG1.4 Input: 1 right answer and 3 wrong answer
VG1.5 Output: "Game Over" and game score
VG1.6 Normal
VG1.7 Blackbox

VG1.8 Functional

VG1.9 Integration

VG1.1 **Vocab Game Test 2**

VG1.2 Ensure invalid inputs are not accepted in game translation input

VG1.3 The game should not accept invalid characters in translation input fields

VG1.4 Input: Translation with invalid characters

VG1.5 Output: "Input is invalid" error message

VG1.6 Abnormal

VG1.7 Blackbox

VG1.8 Functional

VG1.9 Unit

Test Case Matrix

ID	Normal / Abnormal	Blackbox / Whitebox	Functional / Performance	Unit / Integration
UC1.1	Normal	Blackbox	Functional	Integration
UC2.1	Normal	Blackbox	Functional	Unit
UC3.1	Abnormal	Blackbox	Functional	Unit
UC4.1	Abnormal	Blackbox	Functional	Unit
UL1.1	Normal	Blackbox	Functional	Integration
UL2.1	Normal	Blackbox	Functional	Unit
UL3.1	Abnormal	Blackbox	Functional	Unit
SC1.1	Normal	Blackbox	Functional	Integration
SC2.1	Normal	Blackbox	Functional	Unit
SC3.1	Abnormal	Blackbox	Functional	Integration
WA1.1	Normal	Blackbox	Functional	Integration
FC1.1	Normal	Blackbox	Functional	Unit

FC2.1	Normal	Blackbox	Functional	Unit
UD1.1	Normal	Blackbox	Functional	Unit
UD2.1	Normal	Blackbox	Functional	Unit
PP1.1	Normal	Blackbox	Functional	Unit
PP2.1	Abnormal	Blackbox	Functional	Integration
VG1.1	Normal	Blackbox	Functional	Integration
VG2.1	Abnormal	Blackbox	Functional	Unit