

Testing Log

Team 8 - Ebook Perpetual Access Tracking

CS-4820 Software Engineering

By: Beakal Begashaw, Cian Bottomley-Mason,
Dilraj Singh Gill, Ethan Penney, Isaac Wolters, Jasleen Kaur

For: Professor Mohammad Mahdi Hassan

April 2nd, 2024

Integration Testing - Beginning March 25th

Test Case Title: Verify Search Functionality with ISBN Number

Test Case Objective: Ensure that the ebook search functionality correctly returns results when searching by a valid ISBN number.

Preconditions: The application is already open on the main search page and the database contains pre-populated data.

Test Data: ISBN number "9780203994948" for an ebook known to exist in the database.

Test Steps:

- **Step Description:** Enter a known ISBN number into the search field.
 - **Action:** Type "9780203994948" into the ISBN search field and click the "Search" button.
 - **Expected Result:** A search is executed without errors.
- **Step Description:** Verify search results.
 - **Action:** Review the list of search results.
 - **Expected Result:** Ebook with the specified ISBN number is displayed in the results page

Postconditions: The application displays the result on the result page with the entered search displayed on the top

Actual Result (Pass/Fail): Pass

Notes: N/A

Tested By: Beakal Begashaw

Test Case Title: OS Font differences

Test Case Objective: Correct the differences in fonts across Windows and Mac

Preconditions: N/A

Test Data: The application - text fields "Title", "ISBN", "OCN", and "OR"

Test Steps:

- **Step Description:** Change font type
 - **Action:** Change font type in ui files to Arial, a cross-compatible font, from MS shell dlg 2

- **Expected Result:**
- **Step Description:** View change
 - **Action:** Open the application to view font difference
 - **Expected Result:** Font displays as on Windows

Postconditions: Font displays in same way across operating systems

Actual Result (Pass/Fail): Pass

Notes: MS shell dlg 2 is a Windows specific font, so Mac was defaulting to its own font style

Tested By: Isaac Wolters

Test Case Title: Negative Search Results

Test Case Objective: Display unique error messages depending on the negative search result.

Preconditions: The application is already open on the main search page

Test Data: N/A

Test Steps:

- **Step Description:** Search for nothing
 - **Action:** Click search when there are no search items in the search boxes
 - **Expected Result:** Error message displaying, "There are no search items in the search boxes."
- **Step Description:** Perform search with no results
 - **Action:** Perform a search that generates no results, such as "uifeiubfrenwef"
 - **Expected Result:** Error message displaying, "There are no results for the search."

Postconditions: Correct, unique error messages pop up for user.

Actual Result (Pass/Fail): Pass

Notes: N/A

Tested By: Isaac Wolters

Test Case Title: International File Testing

Test Case Objective: Test search functionality across various languages and characters.

Preconditions: The application is already open on the main search page, with the local international test file uploaded.

Test Data: The local international test file, UPEI_Ebooks_local_correct_sample_intl.xlsx

Test Steps:

- **Step Description:** Perform searches
 - **Action:** Copy and paste various ebook titles from the various languages into the search box, testing various languages and special characters.
 - **Expected Result:** The ebook of the copied titles is displayed on the results page.

Postconditions: The ebook of the copied titles is displayed on the results page.

Actual Result (Pass/Fail): Pass

Notes: N/A

Tested By: Isaac Wolters

Test Case Title: Special Character Search Testing

Test Case Objective: Test search functionality across various unusual and special characters.

Preconditions: The application is already open on the main search page.

Test Data: Any CRKN file data or local file data.

Test Steps:

- **Step Description:** Perform searches with special characters
 - **Action:** Perform partial title searches using * with various special characters.
 - **Expected Result:** Only ebooks with that character will be displayed.
- **Step Description:** Perform searches with special characters - specific case with *%*
 - **Action:** Perform partial title searches using *%*
 - **Expected Result:** Only ebooks with % in the title will be displayed.

Postconditions: The ebook of the copied titles is displayed on the results page.

Actual Result (Pass/Fail): Fail

Notes: When testing the % sign, it returned all of the ebooks, similar to just searching *. Looking at the code, this is because we convert * to % to do our partial searches, so the query was just multiple %s. Further testing is needed for all SQLite special characters.

Tested By: Isaac Wolters

Test Case Title: Connection Loss During Updating CRKN Files

Test Case Objective: Test outcome of losing connection to the internet in the middle of updating CRKN files.

Preconditions: The application is in the settings page.

Test Data: New CRKN files at the CRKN website link

Test Steps:

- **Step Description:** Begin updating CRKN files
 - **Action:** Click on “Update CRKN” in settings page and begin updating CRKN files
 - **Expected Result:** Updating process begins
- **Step Description:** Lose connection
 - **Action:** Immediately turn off internet connection, simulating losing internet connection
 - **Expected Result:** Updating is stopped, incomplete changes are rolled back in the database, with appropriate pop-ups informing the user.

Postconditions: Updating is stopped, incomplete changes are rolled back in the database, with appropriate pop-ups informing the user.

Actual Result (Pass/Fail): Fail

Notes: When looking back at the code, it seems we only check for the error of connection loss at the beginning of the scraping. Error handling is needed throughout when requests are made to the URL.

Tested By: Isaac Wolters

Test Case Title: Connection Loss During Updating CRKN Files V2

Test Case Objective: Test outcome of losing connection to the internet in the middle of updating CRKN files.

Preconditions: The application is in the settings page.

Test Data: New CRKN files at the CRKN website link

Test Steps:

- **Step Description:** Begin updating CRKN files
 - **Action:** Begin updating CRKN files
 - **Expected Result:** Updating process begins
- **Step Description:** Lose connection

- **Action:** Immediately turn off internet connection, simulating losing internet connection
- **Expected Result:** Updating is stopped, incomplete changes are rolled back in the database, with appropriate pop-ups informing the user.

Postconditions: Updating is stopped, incomplete changes are rolled back in the database, with appropriate pop-ups informing the user.

Actual Result (Pass/Fail): Pass

Notes: Following changes to the code based on the above test, connection loss was handled and incomplete database transactions were rolled back.

Tested By: Isaac Wolters

Test Case Title: Ensure Export Error Handling

Test Case Objective: Ensure export functionality handles naming errors - no name input, :, and / on MacOS

Preconditions: The application is on the results page with > 0 results displayed.

Test Data: Data displayed in results.

Test Steps:

- **Step Description:** Begin export process
 - **Action:** Click Export on the results page
 - **Expected Result:** export process begins
- **Step Description:** No name test.
 - **Action:** Attempt to save the file with no name input.
 - **Expected Result:** Unable to save file with no name.
- **Step Description:** colon and / test on MacOS
 - **Action:** Attempt to save the file with a colon or / in the name.
 - **Expected Result:** Unable to save file with a colon or / in the name.
- **Step Description:** Invalid characters on Windows
 - **Action:** Attempt to save the file with < > ? / \ | " : *
 - **Expected Result:** Unable to save file with those characters.

Postconditions: Export only allowed if the file name is a valid operating system file name.

Actual Result (Pass/Fail): Pass

Notes: N/A

Tested By: Isaac Wolters, Ethan Penney, Cian

Test Case Title: Ensure CRKN URL Error Handling

Test Case Objective: Ensure CRKN URL input handles naming errors - begins with the protocol (eg. http:// or https://)

Preconditions: The application is on the settings page.

Test Data: https://library.upei.ca and library.upei.ca

Test Steps:

- **Step Description:** Change CRKN URL to library.upei.ca
 - **Action:** Input the URL into the CRKN URL field and click save
 - **Expected Result:** URL is rejected and restore the previous URL.
- **Step Description:** Change CRKN URL to https://library.upei.ca
 - **Action:** Input the URL into the CRKN URL field and click save
 - **Expected Result:** The CRKN URL is changed and updating works.

Postconditions: Only valid URLs are accepted as a CRKN URL.

Actual Result (Pass/Fail): Pass

Notes: N/A

Tested By: Isaac Wolters

Test Case Title: Convert title_metadata_last_modified formatting

Test Case Objective: Convert the column title_metadata_last_modified format to remove the seconds when uploaded to database

Preconditions: The application is on the settings page.

Test Data: N/A

Test Steps:

- **Step Description:** Update CRKN data
 - **Action:** Do a fresh update of CRKN data to ensure the date format gets converted correctly and without errors.
 - **Expected Result:** CRKN data updated as normal.
- **Step Description:** Upload local file.

- **Action:** Upload a local file to ensure the column gets converted for that as well.
- **Expected Result:** Local file uploaded with no issues.
- **Step Description:** Perform search (*)
 - **Action:** Perform a search to see if the column displays in the new, correct format
 - **Expected Result:** Search results display column correctly.
- **Step Description:** Export results.
 - **Action:** Export the results to ensure the column format is correct in the exported file.
 - **Expected Result:** Exported file has the correct format for the column

Postconditions: The title_metadata_last_modified column is in the correct format.

Actual Result (Pass/Fail): Pass

Notes: N/A

Tested By: Isaac Wolters

Test Case Title: Search Error: No search term in search box when 'Search' is clicked

Test Case Objective: Notify user saying there needs to be a search term in the search box to perform a search

Preconditions: The application is on the home searching page.

Test Data: N/A

Test Steps:

- **Step Description:** Update CRKN data
 - **Action:** Do a fresh update of CRKN data to ensure there is data to be searched
 - **Expected Result:** CRKN data updated as normal.
- **Step Description:** Click on Search
 - **Action:** Click on Search without writing any search term in search box
 - **Expected Result:** Pop up info box notifying user that there are no search terms to search

Postconditions: Pop up displayed and application doesn't redirect to the search page

Actual Result (Pass/Fail): Pass

Notes: N/A

Tested By: Beakal Begashaw

Test Case Title: Search Error: No search term in more than one search boxes

Test Case Objective: Notify user saying there needs to be a search term in the first search box

Preconditions: The application is on the home searching page.

Test Data: N/A

Test Steps:

- **Step Description:** Update CRKN data
 - **Action:** Do a fresh update of CRKN data to ensure there is data to be searched
 - **Expected Result:** CRKN data updated as normal.
- **Step Description:** Add one more search box
 - **Action:** Click on + sign to add another search box
 - **Expected Result:** Two empty search boxes appear
- **Step Description:** Click on Search
 - **Action:** Click on Search without writing any search term in the search boxes
 - **Expected Result:** Pop up info box notifying user that there are no search terms in the first search box

Postconditions: Pop up displayed and application doesn't redirect to the search page

Actual Result (Pass/Fail): Pass

Notes: N/A

Tested By: Beakal Begashaw

Test Case Title: Search Error: Valid Search term in one box but not in another

Test Case Objective: Notify user saying there are no search terms in one or more of the search boxes

Preconditions: The application is on the home searching page.

Test Data: N/A

Test Steps:

- **Step Description:** Update CRKN data
 - **Action:** Do a fresh update of CRKN data to ensure there is data to be searched
 - **Expected Result:** CRKN data updated as normal.
- **Step Description:** Add one more search box

- **Action:** Click on + sign to add another search box
- **Expected Result:** Two empty search boxes appear
- **Step Description:** Enter a valid search term in the first box
 - **Action:** Enter a search term leave the second one empty
 - **Expected Result:** First search box contains valid search term; second one doesn't
- **Step Description:** Click on Search
 - **Action:** Click on Search
 - **Expected Result:** Pop up info box notifying user that there are no search terms in one or more of the search boxes.

Postconditions: Pop up displayed and application doesn't redirect to the search page

Actual Result (Pass/Fail): Pass

Notes: N/A

Tested By: Beakal Begashaw

Test Case Title: Search for CRKN data when 'Allow CRKN Data' is turned off

Test Case Objective: Notify user saying there are no search terms when searching for a known CRKN data search term but Allow CRKN is turned off. This functionality is used to only display results from local uploads only.

Preconditions: The application contains CRKN data in database already

Test Data: CRKN data book

Test Steps:

- **Step Description:** Turn off Allow CRKN Data
 - **Action:** Go to settings and turn off Allow CRKN Data
 - **Expected Result:** Allow CRKN is toggled off and user can't Update CRKN data
- **Step Description:** Go to home page and enter a CRKN data search term
 - **Action:** Enter a CRKN data search term and click search
 - **Expected Result:** Pop up appears notifying there are no results for the search.
- **Step Description:** Toggle Allow CRKN back on and perform same search
 - **Action:** Toggle is back on and search term is entered again
 - **Expected Result:** Application shows the results for that term in the results page

Postconditions: Application shows the search results

Actual Result (Pass/Fail): Pass

Notes: N/A

Tested By: Beakal Begashaw

Unit Testing

Database

Test Case Name: test_connect_to_database

- **Test Description:** This test verifies that the `connect_to_database` function successfully connects to the SQLite database using the settings manager to retrieve the database filename. It mocks the `Settings.get_setting` method to return a predefined database name ('test_database.db') and asserts that `sqlite3.connect` is called with this database name.
- **Expected Result:** The function should call `sqlite3.connect` with 'test_database.db' as its argument and return a mock connection object.
- **Actual Result (Pass/Fail):** Pass
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Isaac Wolters, Cian
- **Code Reviewer's Name:** Isaac Wolters
- **Comments/Notes:** N/A

Test Case ID: test_close_database

- **Test Description:** This test verifies that the `close_database` function in the `database` module correctly commits any pending transactions and closes the database connection.
- **Expected Result:** The `close_database` function should call the `commit` method to save any changes made during the connection session and then call the `close` method to properly close the connection to the SQLite database. Both methods are expected to be called exactly once.
- **Actual Result (Pass/Fail):** Pass
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Cian, Isaac Wolters
- **Code Reviewer's Name:** Isaac Wolters
- **Comments/Notes:** N/A.

Test Case Name: test_get_CRKN_tables_allow_true

- **Test Description:** Test the functionality of getting the list of CRKN table names if allow_CRKN is set to True

- **Expected Result:** The function should retrieve a list of CRKN table names or empty if allow_CRKN is turned off by the user.
- **Actual Result(Pass/Fail):** Pass
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Isaac Wolters
- **Code Reviewer's Name:** Beakal Begashaw
- **Comments/Notes:** N/A

Test Case Name: test_get_local_tables

- **Test Description:** Tests the functionality of getting the table names from local
- **Expected Result:** Returns list of all local table names
- **Actual Result (Pass/Fail):** Pass
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Isaac Wolters
- **Code Reviewer's Name:** Beakal Begashaw, Isaac Wolters
- **Comments/Notes:** N/A

Test Case Name: test_get_tables_integration

- **Test Description:** Tests the functionality of getting the combined names of local and CRKN table names by making use of two helper functions.
- **Expected Result:** list of all table names; both local and CRKN table names
- **Actual Result (Pass/Fail):** Pass
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Isaac Wolters
- **Code Reviewer's Name:** Beakal Begashaw, Ethan Penney
- **Comments/Notes:** N/A

Test Case Name: test_tables_need_creation

- **Test Description:** Test the scenario where both tables need to be created if they don't exist already
- **Expected Result:** creation of new tables in database
- **Actual Result (Pass/Fail):** Pass
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Isaac Wolters
- **Code Reviewer's Name:** Beakal Begashaw, Ethan Penney
- **Comments/Notes:** N/A

Test Case Name: test_tables_already_exist

- **Test Description:** Test the scenario where both tables already exist
- **Expected Result:** no new creation of tables in the database
- **Actual Result (Pass/Fail):** Pass

- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Isaac Wolters
- **Code Reviewer's Name:** Beakal Begashaw, Ethan Penney
- **Comments/Notes:** N/A

Test Case Name: test_search_by_title_with_wildcards

- **Test Description:** Test search functionality with wildcards
- **Expected Result:** result should return the string of the search term entered
- **Actual Result (Pass/Fail):** Pass
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Isaac Wolters
- **Code Reviewer's Name:** Cian
- **Comments/Notes:** N/A

Test Case Name: test_get_table_data

- **Test Description:** Test the functionality of retrieving data from a specific table in the database
- **Expected Result:** result should return data from any table in the database
- **Actual Result (Pass/Fail):** Pass
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Ethan Penney
- **Code Reviewer's Name:** Isaac Wolters
- **Comments/Notes:** N/A

Scraping

Test Case Name: test_scrapeCRKN_success

- **Test Description:** Test the functionality of a successful scrape of an HTML content with xlsx files.
- **Expected Result:** the two xlsx files in the mock HTML content should be retrieved and processed without any errors
- **Actual Result (Pass/Fail):** Pass
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Isaac Wolters
- **Code Reviewer's Name:** Beakal Begashaw, Ethan Penney
- **Comments/Notes:** N/A

Test Case Name: test_scrapeCRKN_fail_http_error

- **Test Description:** Test the functionality of a failed scrape of an HTML content with xlsx files by mocking an HTTP error when get request is made.

- **Expected Result:** the two xlsx files in the mock HTML content should not be processed and an error should be thrown
- **Actual Result (Pass/Fail):** Pass
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Isaac Wolters, Ethan Penney
- **Code Reviewer's Name:** Beakal Begashaw, Ethan Penney, Isaac Wolters
- **Comments/Notes:** N/A

Test Case Name: test_compare_file_insert

- **Test Description:** Test the functionality of a comparing the recently scraped file to check if it is already in the database. This test function should insert into the database.
- **Expected Result:** should run an 'INSERT INTO' command to insert the new file in the database
- **Actual Result (Pass/Fail):** Pass
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Isaac Wolters
- **Code Reviewer's Name:** Beakal Begashaw, Ethan Penney, Isaac Wolters
- **Comments/Notes:** N/A

Test Case Name: test_compare_file_update

- **Test Description:** Test the functionality of a comparing the recently scraped file to check if a file in the database needs to be updated with the newly scraped file
- **Expected Result:** should run an 'UPDATE' command to replace the old file with the new file.
- **Actual Result (Pass/Fail):** Pass
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Isaac Wolters
- **Code Reviewer's Name:** Beakal Begashaw, Ethan Penney, Isaac Wolters
- **Comments/Notes:** N/A

Test Case Name: test_update_tables_insert

- **Test Description:** Test the functionality of updating CRKN or local tables with file information that has been scraped(publisher, date/version number...)
- **Expected Result:** should run the INSERT INTO command with the new file information that has been scraped into already existing tables
- **Actual Result (Pass/Fail):** Pass
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Isaac Wolters
- **Code Reviewer's Name:** Beakal Begashaw, Ethan Penney, Isaac Wolters
- **Comments/Notes:** N/A

Test Case Name: test_update_tables_update

- **Test Description:** Test the functionality of updating CRKN or local tables with file information that has been scraped(publisher, date/version number...) by updating with the new one
- **Expected Result:** should run the UPDATE command with the new file information
- **Actual Result (Pass/Fail): Pass**
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Isaac Wolters
- **Code Reviewer's Name:** Beakal Begashaw, Ethan Penney, Isaac Wolters
- **Comments/Notes:** N/A

Test Case Name: test_check_file_format_correct

- **Test Description:** Test the functionality of checking the file format of the xlsx file to make sure all the headers are in the right order
- **Expected Result:** should return true based on mock correct file format
- **Actual Result (Pass/Fail): Pass**
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Isaac Wolters
- **Code Reviewer's Name:** Ethan Penney, Isaac Wolters
- **Comments/Notes:** N/A

Test Case Name: test_check_file_format_incorrect_header

- **Test Description:** Test the functionality of checking the file format of the xlsx file to make sure all the headers are in the right order
- **Expected Result:** should return false based on mock incorrect file format
- **Actual Result (Pass/Fail): Pass**
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Isaac Wolters
- **Code Reviewer's Name:** Ethan Penney, Isaac Wolters
- **Comments/Notes:** N/A

Settings Manager

Test Case Name: test_singleton_behavior

- **Test Description:** Test the functionality of a singleton instance being created when settings_manager is called so that different modules can get different informations from settings.json.

- **Expected Result:** Two instances should point to the same settings_manager instance in memory
- **Actual Result (Pass/Fail):** Pass
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Beakal Begashaw
- **Code Reviewer's Name:** Isaac Wolters
- **Comments/Notes:** N/A

Test Case Name: test_load_settings_successfully

- **Test Description:** Test the functionality of retrieving the correct value for a key when get_setting is called.
- **Expected Result:** Correct value is retrieved.
- **Actual Result (Pass/Fail):** Pass
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Beakal Begashaw
- **Code Reviewer's Name:** Isaac Wolters
- **Comments/Notes:** N/A

Test Case Name: test_handle_missing_settings_json

- **Test Description:** Test the functionality of creating a settings.json file if it is not there
- **Expected Result:** If settings.json is not available, create the json file with default settings
- **Actual Result (Pass/Fail):** Pass
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Beakal Begashaw
- **Code Reviewer's Name:** Isaac Wolters
- **Comments/Notes:** N/A

Test Case Name: test_update_setting

- **Test Description:** Test the functionality of updating a key's value such as language
- **Expected Result:** The new value should be loaded in the settings.json file
- **Actual Result (Pass/Fail):** Pass
- **Tester's Name:** Beakal Begashaw
- **Code Generator's Name:** Beakal Begashaw
- **Code Reviewer's Name:** Isaac Wolters
- **Comments/Notes:** N/A

