PROG 3B POE

Documentation

Application: Dewey Training
Application ID: DT1.20

Version: 1.20
Student Name: Karl Dicks
Student Number: 17667327
Course: BCAD3
Subject: PROG7312
Lecturer: Nirasha Ramckurran
Assignment: POE
Due Date: 04/11/2020

Contents

Change Log 2
Introduction 2
Task 1 Improvements 5
Task 2 Improvements 6
Conclusion 7
Introduction 8
Help File 10
Readme 16
Screenshots 20
Database Entities 40
Tree Data Structure 41
Use Case Diagram 42
Conclusion 43
References 44

Change Log

Introduction

As part of our PROG7312 (3B) module, we were tasked with developing a Dewey Decimal training application. My application has been built in Windows Presentation Foundation (WPF) in .Net Core 3.1. There were a number of requirements that the application had to perform, which have been implemented in my Task 1, Task 2, and POE project, in accordance with our set question paper.

Task 1 consisted of two parts: research into gamification features to be implemented in the practical part of the assignment, and the practical part of the assignment, being the reordering of Dewey Decimal books on a virtual shelf.

The requirements of the practical part of Task 1 is described below:

1. On start-up, the application shall allow the user to choose between three tasks:
    a. Replacing books.
    b. Identifying areas.
    c. Finding call numbers.
2. For this first task, only Replacing books will be implemented – disable the other two options for now.
3. When the user selects Replacing books, the application shall randomly generate ten different call numbers, and display them to the user.
4. The application shall allow the user to reorder the call numbers, and the application shall check whether the user got the ordering right.
5. Implement the gamification feature that you identified to motivate users to keep learning.

Technical requirements:

1. Make use of a list to store the generated call numbers.
2. Choose an appropriate sorting algorithm to sort the call numbers to check the order that the user put them in.

These criteria were met for my Task 1 submission, and I have received feedback for this task, as detailed in the following section.

Task 2 only had an implementation component to it, which had the following functions / criteria:

1. Enable the Identifying areas task.
2. When the user chooses the Identifying areas task, they should be presented with a user interface where they will match two columns: call number (top level only) and description.
3. A question in this context is defined as the whole matching set, including both columns.
4. The user shall be allowed to answer as many questions as they want to.

5. The questions should alternate between matching descriptions to call numbers and call numbers to descriptions.
6. Each question should have four randomly selected items in the first column, and seven possible answers (three of which are incorrect) in the second column.
7. Implement a gamification feature to motivate users to keep using the application. You may use the same one as before or choose to implement a different one.

Technical requirements:

1. Store the call numbers and their descriptions in a dictionary.

These criteria were met for my Task 2 submission, and I have received feedback for this task, as detailed in the following section.

For our POE submission, we were tasked with improving our Task 1 and 2 submission, along with completing theory and practical parts as follows:

In a Word document, create a multilevel list showing the call numbers, for example:

- 700 Arts & Recreation
    - 750 Paining
        - 751 Techniques, procedures, apparatus, equipment, materials, forms
        - 752 Color

This was completed for 160 unique entries, and has been submitted in my Portfolio of Evidence. The references for this information were included.

The practical component of this assignment has been detailed below:

2. Create a file containing the data that was gathered in the research part of this task in a format that can be read by your application.
3. Enable the Finding call numbers task.
4. When the user chooses Finding call numbers, the application must load the Dewey Decimal classification data into memory from the file created in step 1.
5. The quiz must work as follows:
    a. For each question, randomly select a third level entry from the data, for example 752 Color. Display only the description, not the call number.
    b. Display four top level options to the user to choose between, one of which must be the correct one and the other three randomly selected incorrect answers. For example:
    000 General Knowledge
    400 Language
    700 Arts & Recreation
    800 Literature
    c. For the options, display both the call number and description. Display the options in numerical order by call number.
    d. If the user selects the correct option, show them four options from the next level, until the most detailed level is reached.

e. If the user selects the wrong option anywhere along the way, indicate this and then ask the next question.

Technical requirements:

1. Make use of a tree to store the data in memory.

These criteria have been met, by implementing the complete solution for the POE submission, as well as the research document.

Certain aspects of my Task 1 and 2 were improved, and have been detailed in the next section, in accordance with feedback received for both Task 1 and Task 2.

Task 1 Improvements

The following feedback was received for my Task 1 submission:

Dicks, Karl		
	Max	***Student***
Research: List of at least five gamification features.	5	5
Research: Introduction and conclusion.	5	3
Research: Motivation for choice of gamification feature.	10	8
App Functionality: Allow the user to choose which task to	10	10
App Functionality: Display ten randomly generated call	10	10
App Functionality: User can change the order of the call	10	10
App Functionality: App checks whether the user got the ordering correct.	10	10
App Functionality: A gamification feature is implemented.	10	9
App Logic: A list is used to store the call numbers.	5	5
App Logic: An appropriate sorting algorithm is used to sort the call numbers.	10	10
Coding Standards: Code is well structured and documented.	5	4
Documentation: Readme file provides enough information to run the app.	5	4
Other marks: App is easy to use	5	4
Other Marks: Advanced features not covered in class (Bonus Marks).	5	0
Total	100	92
Penalty		0
Percentage		***92***

Comments
A well presented program. Well researched. Nice choice of gamification features. Concepts are explained well in the research doc. Readme file added and done well. Demo done via youtube link. Demo presented well with a voice over. Timer added. Good use of visualisation for the ordering of books. Program is fun nice use of colors and themes. A very good knowledge of programming concepts displayed. nice use of the DAL. Improve the conclusion and in text refs. Add a few more gamification concepts in theory that you think may also work.

I have made the following changes to my Task 1 assignment:

- Introduction and conclusion have been improved in my Task 1 assignment.
- Motivation for choice of gamification feature has been expanded upon.
- Additional gamification features have been included in my Task 1 research.
- Gamification feature implementation has been improved by making it more clear to the user (background included for the timer).
- Coding standards improved by including additional comments.
- Application ease of use improved.
- Readme updated.
- Grammar corrected.
- Spelling mistakes corrected.
- In text references updated, for additional information included.
- Reference list updated.

The above changes have been made to my Task 1 assignment, in order to update its content and improve it based on lecturer feedback.

Task 2 Improvements

The following changes have been made to my Task 2 assignment, in order to improve on the original design, and functionality for the POE submission:

The following feedback was received for my Task 2 submission:

Dicks, Karl			
	Max	***Student***	***Comments***
App Functionality: User can choose the new Identifying areas task.	5	5	Very well put together program well researched and presented logins done although not required. Colours and themes set nicely. Nice dragging feature to order the books. Some menu options seem repeated. Readme file done well. Demo video done via YouTube. Link provided in readme file. Timing and scoring added. Identify the areas done well. Questions are randomised. Matching of columns work well. Program score as well as times as you. Program also gives the correct answers if you don't know them - nicely done. Scoreboard added. Readme doesnt have anything abt the latest feature. Nice use of DAL & doubly linked lists. Changelog?
App Functionality: User is presented with a randomly generated match-the-columns question, with more answers than questions.	20	15	
App Functionality: The questions should alternate between descriptions to call numbers and call numbers to descriptions.	10	10	
App Functionality: User can complete the match the columns question.	10	10	
App Functionality: App checks whether the selected answers are correct	10	10	
App Functionality: App allows the user to keep practising.	10	10	
App Functionality: Gamification feature implemented.	10	10	
App Logic: Data stored in a dictionary.	10	10	
Coding Standards: Code is well structured and documented.	5	5	
Documentation: Readme	5	3	
Other marks: App is easy to use.	5	4	
Other Marks: Advanced features not covered in class (Bonus Mark	5	0	
Total	100	92	
Referencing penalty		0	
Final mark		92	

- Readme updated.
- Application ease of use improved (timer easier to see).
- Coding standards improved by including additional comments and breaking the application into more methods.
- Changelog included in the final POE.

The above changes have been made to my Task 2 assignment, in order to update its design and functionality based on lecturer feedback.

Conclusion

In conclusion, the above-mentioned changes have been implemented in my Task 1 and Task 2 assignments, in order to improve their quality. Additional design improvements have been included in the practical submission, in order to improve its design, and make it look more professional and attractive for its users.

Debugging has been carried out, in order to address application bugs and issues which were not obvious in Task 2. These issues have since been resolved, and changes have been made for the final POE submission.

In addition to the above, the POE has included additional functionality as set out by the question paper, and the research has been completed.

Introduction

As part of our PROG7312 (3B) module, we were tasked with developing a Dewey Decimal training application. I chose to develop the application in C#, using the Visual Studio 2019 IDE, as we were familiar with this IDE from other programming modules. My application has been built in Windows Presentation Foundation (WPF) in .Net Core 3.1. There were a number of requirements that the application had to perform, which have been implemented in my Task 1, Task 2, and POE project, in accordance with our set question paper. (The Independent Institute of Education, 2020)

The training application allows users to perform multiple actions, including:

- **Register and login**
  The training application allows the user to register an account with their own preferences, and log in. User profiles are stored in a local MSSQL MDF file, along with their scores used by the application. This database can easily be migrated to an online Azure database, so that high scores are accessible by everyone in the library or elsewhere.

- **Replace Books**
  The application will allow users to order randomly generated Dewey Decimal Call Numbers (10) – including the decimals and authors into the correct order.
  Once the user correctly orders the call numbers, by dragging the books to their correct order, the user will be automatically navigated to a confirmation page.

- **Identify Areas**
  The application allows users to complete match-the-column questions with 4 questions being asked, and 7 potential answers. These questions make up the full Dewey Decimal area / answer – e.g. 000 – General Knowledge. The user may select answers to the questions from dropdown menus, and will determine score, time taken, and other such information when the “Next” button is clicked.

- **Find Call Numbers**
  The application will allow users to find call numbers by implementing a tree structure, providing the user with a third level entry description – e.g. Color, and then allowing the user to drill down from first, to second, to third level Dewey Decimal. For example, 700 Arts & Recreation -> 750 Painting -> 752 Color.

- **Gamification Techniques**
  A number of gamification techniques have been implemented, including all those described in the research document.
  These include:
    - **Leaderboards**
      The user will be able to see the top ten scores (game completion times), as these are saved for signed-in users and stored in the database file. These scores are then retrieved and displayed on the home screen when the user first loads the application. (Quicksprout, 2016)
    - **Challenges**
      The application has implemented a timer, and difficulty levels. For example, the user can set the difficulty to "Easy" which allows the user 60 seconds to complete the ordering / test process. They can set it to "Medium" for 40 seconds, and "Hard" for 30 seconds. This provides different levels of difficulty for the user to complete the ordering in set timeframes. (Quicksprout, 2016)
    - **Feedback**
      The application displays a timer, and once the timer reaches 10 seconds, it will start to alternate between red and white text color, to indicate that the time is almost finished for the user to complete the ordering. (Laja, n.d.)

    - **Rewards**
      The application shows the top ten scores on the home screen, and these scores are for logged in users only, as the score is linked to their user account. Anonymous users can still use the application without logging in, and will receive their time, however it will not be logged and displayed on the home screen.
      The top three scores will have different colours, much like a podium system, where the top scorer gets their row in gold, second in silver, and third in bronze. (Laja, n.d.)
    - **Progress**
      In addition to the above gamification techniques, the user will be able to view all their personal scores on a grid view, and this will be displayed by highest score first (lowest time taken to complete the ordering). This allows the user to track their progress over time, if they are logged in. (Quicksprout, 2016)

- **Restart Training**
  Finally, the user will be able to restart their game by pressing the "Restart" button on the "Replace Books" page. This will reset the timer, replace the books with new auto-generated call numbers and authors, and allow them to start the game again. (The Independent Institute of Education, 2020)

Help File

The training application provides numerous functions, which will be described in depth in the following section. The help file has been broken up into multiple sections, describing each page of the desktop application.

Home Page

The application will first load to the home page, as an "Anonymous User". An anonymous user can view all high scores, set difficulty of the game, login, register, and access the "Replace Books" function required for Task 1.

- The high scores will be displayed in a data grid on the right-hand side of the home screen, with scores (time taken to complete the ordering of the books, or categorize the books), usernames, and the date and time that the user took the test.

- The user can access the "Replace Books", "Identify Areas", or "Find Call Numbers" functions, however they will not be able to save their high scores or access their score log without signing in. Therefore, their high score will not be displayed on the home screen.

- In addition to this, the top three scores are displayed with different background colours, in order to create a podium-like system. For example, the highest score will be displayed with a gold background, second highest will be displayed with a silver background, and the third will be displayed with a bronze background.

- The user may use button click events to complete the following actions:
    - Navigate to the login / registration page,
    - Navigate to the replace books page,
    - Navigate to the identify areas page,
    - Navigate to the view all scores page,
    - Exit the training application,
    - Minimize the window.

- Once the user has logged in, and is no longer an anonymous user, they may select a "View All Scores" button on the home screen, which will display their score log on a data grid.

- The user may select the different score views by selecting the "Set Score View" combo box and choosing a test type.

- The user may also set the game difficulty by selecting the combo box at the bottom of the main page, and from there they can select either "Easy", "Medium", or "Hard", which will set the time limit for the various games.

The login functionality will be described next.

Login Page
Once the user selects the “Login” button on the home screen, they will be brought to a page where they can log into an existing account, or register a new account on the system.

The login page provides a number of buttons, and input, including:

- Username input,
- Password input,
- Register button – brings the user to a registration page,
- Login button – logs the user in,
- Back button (left arrow at the top left of the screen).

The user can enter their unique credentials, and log into their account. The logged in username at the top left of the screen will be changed to “User: <username>”, e.g. “User: Karl”.

Once the user has logged in successfully, they will be brought back to the home page, where they can navigate to all functionality of the application.

The login page validates user accounts, and therefore if the user credentials are incorrect, they will not be logged in, and a generic “Username / Password Incorrect” message will be displayed.
This is a generic message, as to not encourage username harvesting, whereby users attempt to determine valid usernames, if only the password is incorrect – e.g. the message does not display “Password Incorrect”, when a username is in fact valid, however a password is incorrect.

Register Page
The user can navigate to the register page, once they press the “Register” button on the login page. This page allows new users to sign up an account with the system.

A number of buttons and input boxes are displayed on the register page, including:

- Username input,
- Password input,
- Confirm password input,
- Register Account button,
- Back button (left arrow at the top left of the screen).

The user can enter unique credentials on this page, and create a new account with the system. If the user enters a username already in use, an error message will be displayed, and password validation ensures both the confirm password and password input boxes contain the same password values.

Once the user enters valid credentials, they can press the “Register Account” button, which will create the account in the database, and navigate the user back to the login page, where they can login with their new details.

Replace Books
This page provides functionality required for replacing books, where the user can:

- View auto-generated Dewey Decimal Call Numbers, which are shown on a data grid on the right of the page. These Dewey Decimal's show the decimal as well as the author.
  Both of these values (decimal, and author) are automatically generated using a custom random generator.

- Re-order the books into their correct order, as the application randomly places books on the "shelf", for example, the Dewey Decimal system requires that all books are ordered by numerical order as well as alphabetically – e.g. "035.8605 NVG" comes before "035.8605 ZAK" or "035.8605 NVG" comes before "125.8605 ABC".
  The user can simply drag and drop rows (books) into the correct order on the "shelf", once they press the "Start" button.

- Start the game by pressing the "Start" button on the bottom left of the page. Once started, the timer will start to tick down to show the time remaining. The difficulty setting will determine how long the user has to correctly order the books.
  In addition to this, the Call Numbers will be generated again, so users cannot determine where the books should be before starting the game, and have the timer tick down.

- The user may return to the main page by pressing the "Return" button. This will cancel the current training session or "game". This score will not be logged, and the user will have to restart the game to start again.

Once the user re-orders all books, they will be navigated to a confirmation page, where they can view time taken to order the books (score in seconds). If they are logged in, this score will be saved in the database, along with the logged in user and time taken of the test.

Replace Books Confirmation Page
Once the books have been ordered, or the time has run out, the user will be navigated to a confirmation page, which will either display "The books have been successfully ordered" or "The books have not been successfully ordered".
When the user orders the books in the correct order within the specified timeframe (e.g. 30 seconds), the time taken to complete the ordering of the books will be displayed.

In addition to showing this information, the following actions can be completed on this page:

- Finish – navigates the user back to the home page.
- View Order – navigates the user to a page where the correct order of the books is displayed, according to the Dewey Decimal System – in numerical and alphabetical order.

Identify Areas
This page provides functionality required for identifying areas, where the user can:

- View match the column questions, which have top level (category) call numbers and their descriptions in two list boxes. One list box has the questions, which can contain both descriptions or top-level call numbers. There are 4 questions, and 7 possible answers (4 of which are correct), and are displayed in the second list box.

- The user may start the game, which will allow them to select items from the drop-down lists. Once the user presses the “Start” button, they can select a possible answer from the right-hand column by selecting the letter associated with the answer (e.g. A. General Knowledge). The questions and answers will be automatically shuffled once the user starts a new game, however there will be 4 correct answers to the questions within the second list box.

- Once all answers have been selected from the drop-down lists, the user may proceed to click “Next”, which will check their answers against the model answer. It will navigate the user to a confirmation page that will show the time taken to complete the test, how many questions were correctly answered, and also whether the test was completed successfully (whether all answers were correct). This information will be stored in a database, which is shown on the leaderboard.

- The user can continue to play another game, with another set of questions and randomly generated answers (with 4 being correct), view the model answer, or finish the game. On the confirmation page their score will be presented, along with the time taken to complete the test.

Once the user correctly assigns the descriptions to the categories or vice versa, they will be navigated to the confirmation page described above. If they are logged in, this score will be saved in the database, along with the logged in user, score, and time taken of the test.

Identify Areas Confirmation Page
Once the descriptions have been categorized, or the time has run out, the user will be navigated to a confirmation page, which will either display a successful message or unsuccessful message.

In addition to showing this information, the following actions can be completed on this page:

- Finish – navigates the user back to the home page.
- View Answers – navigates the user to a page where the correct answers can be viewed. If the user has not categorized the top-level call numbers correctly, they can then view this model answer after the test has been completed.
- Next – Navigates the user to a new test, so they can continue to play the game until such time as they press the “Finish” button.

Find Call Numbers
This page provides functionality required for finding call numbers, where the user can:

- View the third level description – e.g. Color, and a set of three incorrect first level call numbers and descriptions, and a further correct call number and description – in this case 700 Arts & Recreation. These four items are displayed in a list box, with their call number followed by their description. A timer will start to tick down from 60, 40, or 30 seconds based on the difficulty setting (which can be set on the home page).

- The game will automatically start when the user presses the "Find Call Numbers" button on the home screen, which will tick the timer down from the defined time limit (according to the difficulty setting, either 60, 30, or 20 seconds). The user may select their answer (which at first will be the first level call number and description), and then click "Next". If their selection was correct, it will allow them to continue to select the second level call number and description, else it will redirect the user to a page which will show a message indicating that the answer was incorrect. They may click "Next" to continue to a new question, view the correct answers (model answer), or click "Finish" to navigate back to the main menu.
  If the question was correctly answered (the top-level Dewey Decimal was selected), then the user may continue to select the second and then third level answer, just as they did for the first question. The questions in the list box will be changed to the second level and third level entries as the user progresses through the test.

- Once all answers have been selected from the list box, the user may proceed to click "Finish", which will check their answers against the model answer. It will navigate the user to a confirmation page that will show the time taken to complete the test, how many questions were correctly answered, and also whether the test was completed successfully (whether all answers were correct). This information will be stored in a database, which is shown on the leaderboard.

- The user can continue to play another game, with another set of questions (call numbers / descriptions) and randomly generated answers (with 3 being correct), view the model answer, or finish the game. On the confirmation page their score will be presented, along with the time taken to complete the test.

Once the user correctly selects all level one, two, and three call numbers and descriptions, they will be navigated to the confirmation page described above. If they are logged in, this score will be saved in the database, along with the logged in user, score, and time taken of the test.

Find Call Numbers Confirmation Page
Once all levels have been selected correctly, or the time has run out, the user will be navigated to a confirmation page, which will either display a successful message or unsuccessful message.

In addition to showing this information, the following actions can be completed on this page:

- Finish – navigates the user back to the home page.
- View Answers – navigates the user to a page where the correct answers can be viewed. If the user has not selected any level answers correctly, then they can view this model answer after the test has been completed.
- Next – Navigates the user to a new test, so they can continue to play the game until such time as they press the “Finish” button.

Readme

Project Title: Dewey Training

Welcome to Dewey Training. This desktop application has been developed for librarians and other users to learn the Dewey Decimal ordering system. The aim of this application is to get librarians and other users of the system to order and manage books efficiently at libraries. This will improve efficiency, and accuracy of these users, when they replace books on the numerous shelves in a library. In addition to this, the categories of these books can be learnt by taking “Identifying Areas” tests on the application. The application has also made provision for finding specific books on shelves, by implementing a tree structure where a certain number of book categories have been collected and populated into the application, and the user must select the correct call numbers / descriptions for level 1, 2, and 3 of these books by “drilling down” from level 1 through 3.

This application encourages users to improve their book replacement and categorization efficiency, and by extension the learning of the Dewey Decimal system. By implementing gamification techniques, such as leaderboards, challenges, feedback, rewards, and progress, the user is encouraged to compete with one another, and learn in the process. By tracking and displaying this information, the user is more likely to see the training software as a game, and therefore compete with one another.

In addition to this, the database will eventually be deployed to an online hosting platform (e.g. Azure SQL database), so that users on different devices may be able to compete with one another – they will be able to see one single leaderboard across multiple devices.

Getting Started

The following steps are required to get the Dewey Training software running on the development environment:

- Open the application source code in Visual Studio
- Set the start-up project to “Dewey Training”
- Run the application on any Windows PC
- Ensure the system is using the dot “.” Delimiter for decimals (EN-US)

Prerequisites

There are a few prerequisites required to run the application, including:

- Install the *latest Visual Studio
- Install prerequisites to run .Net Core 3.1 WPF desktop applications

*latest Visual Studio as of when the application was developed is: Visual Studio 2019

More detailed specifications are included below

Microsoft Visual Studio Enterprise 2019
Version 16.7.2
VisualStudio.16.Release/16.7.2+30413.136
Microsoft .NET Framework
Version 4.8.04084

Installing

- Open the application source code in Visual Studio
- Set the start-up project to “Dewey Training”
- Run the application on any Windows PC

The development test system has been detailed on the following page.

Test System
Development PC

OS Name	Microsoft Windows 10 Pro
Version	10.0.19041 Build 19041
Other OS Description	Not Available
OS Manufacturer	Microsoft Corporation
System Name	KARL
System Manufacturer	System manufacturer
System Model	System Product Name
System Type	x64-based PC
System SKU	SKU
Processor	Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz, 3696 Mhz, 6 Core(s), 12 Logical Processor(s)
BIOS Version/Date	American Megatrends Inc. 2402, 2020/06/17
SMBIOS Version	3.0
Embedded Controller Version	255.255
BIOS Mode	UEFI
BaseBoard Manufacturer	ASUSTeK COMPUTER INC.
BaseBoard Product	ROG MAXIMUS X HERO
BaseBoard Version	Rev 1.xx
Platform Role	Desktop
Secure Boot State	Off
PCR7 Configuration	Binding Not Possible
Windows Directory	C:\WINDOWS
System Directory	C:\WINDOWS\system32
Boot Device	\Device\HarddiskVolume7
Locale	United States
Hardware Abstraction Layer	Version = "10.0.19041.423"
User Name	KARL\Karl
Time Zone	South Africa Standard Time
Installed Physical Memory (RAM)	48.0 GB
Total Physical Memory	47.9 GB
Available Physical Memory	35.0 GB
Total Virtual Memory	54.9 GB
Available Virtual Memory	37.1 GB
Page File Space	7.00 GB
Page File	C:\pagefile.sys
Kernel DMA Protection	Off
Virtualization-based security	Not enabled

Hyper-V - VM Monitor Mode Extensions	Yes
Hyper-V - Second Level Address Translation Extensions	Yes
Hyper-V - Virtualization Enabled in Firmware	Yes
Hyper-V - Data Execution Protection	Yes

Built With
Visual Studio – The IDE used to develop the desktop application
.NET Core 3.1 – Framework
WPF – Windows Presentation Foundation – Used to design the application in C# and XAML
Models – Used to structure data within the application.
Data Access Layer (DAL) – Assembly used to access the database.

Versioning

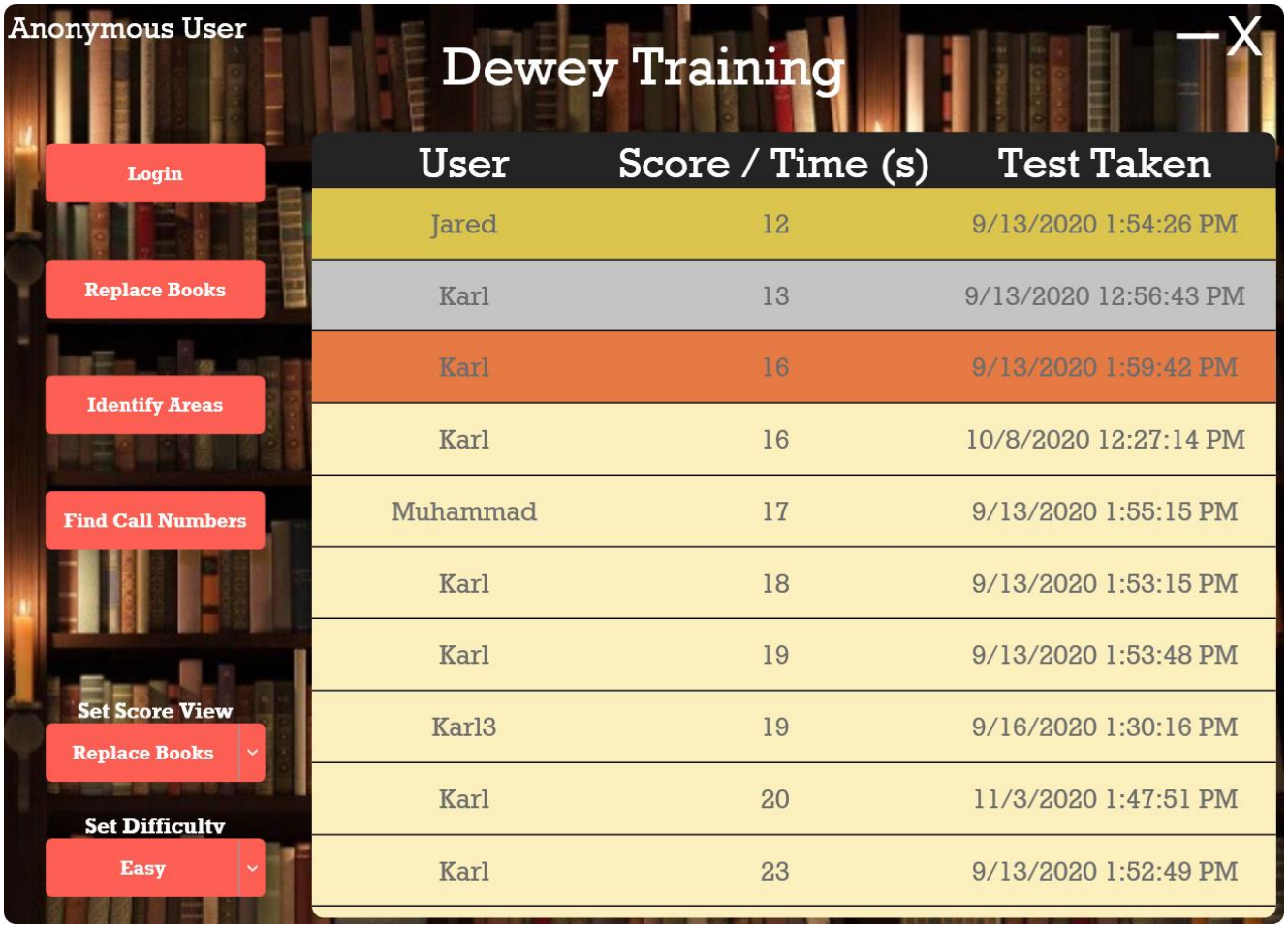
Version 20

Authors Karl Dicks – 17667327
Acknowledgments Inspiration: Programming 3B POE Question Paper
Demo Video link: https://youtu.be/Y-A65WPcPGM

Screenshots

The user interface for Dewey Training desktop application has been designed, and all functionality has been implemented. Below is the interface for my application:

Home Page



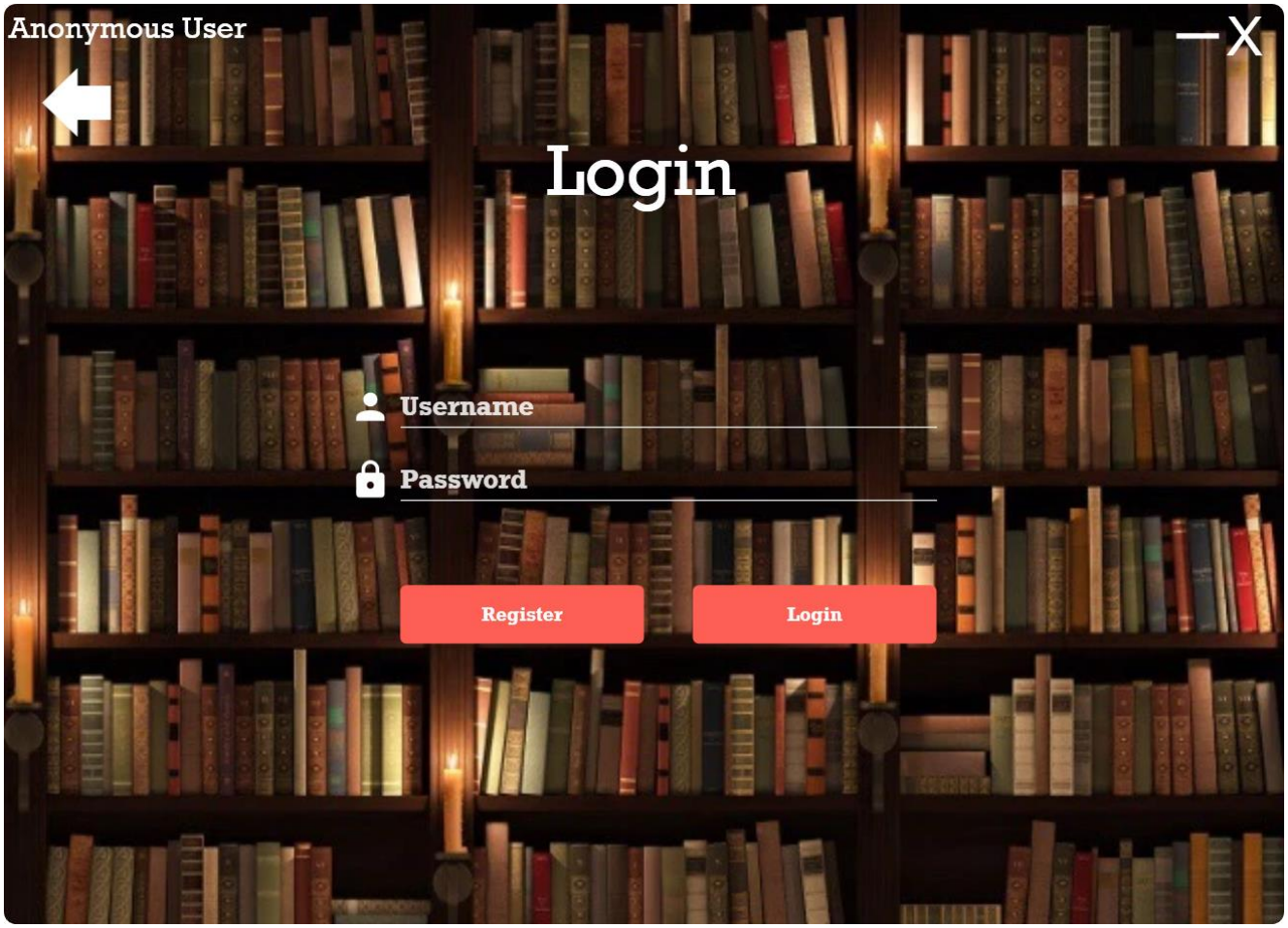
Once the user loads the application for the first time, they will be presented with the home screen, and will not be logged in.

The user can navigate to the login page, or complete training “games” anonymously, which will not save their scores to the database.

If the user would like to log in, and save their scores to the database, they can log into their account by pressing the “Login” button.

This action will bring them to the page provided on the following page.

Login



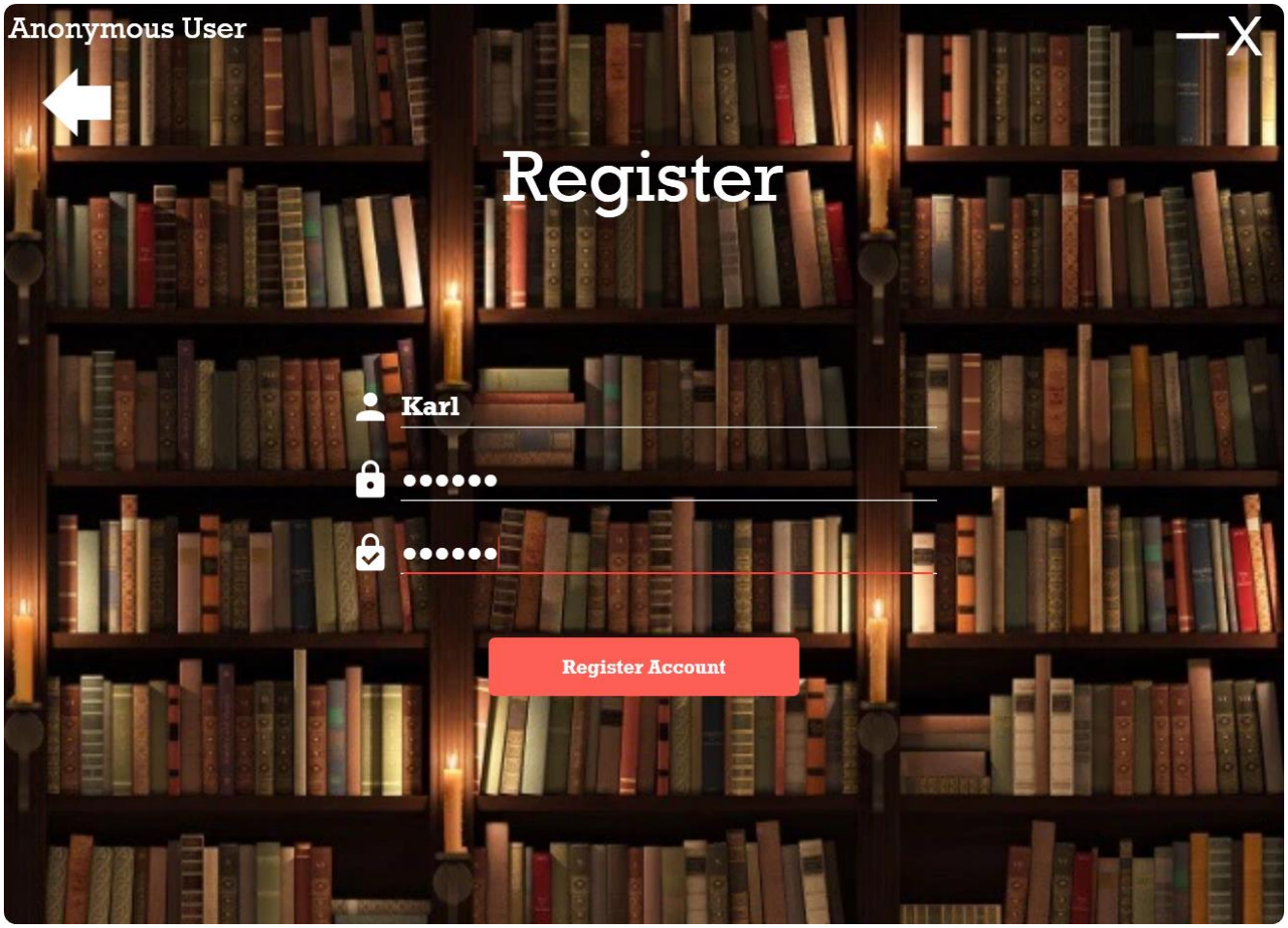
Once the user navigates to the login page, they can either log in with their previously created account, or register a new account by pressing the "Register" button.

Once the user registers a new account, they are brought back to the login page, and can enter their account details.

Once the user has pressed login, and the account is valid, they are brought back to the home page.

The register page is shown on the following page.

Register

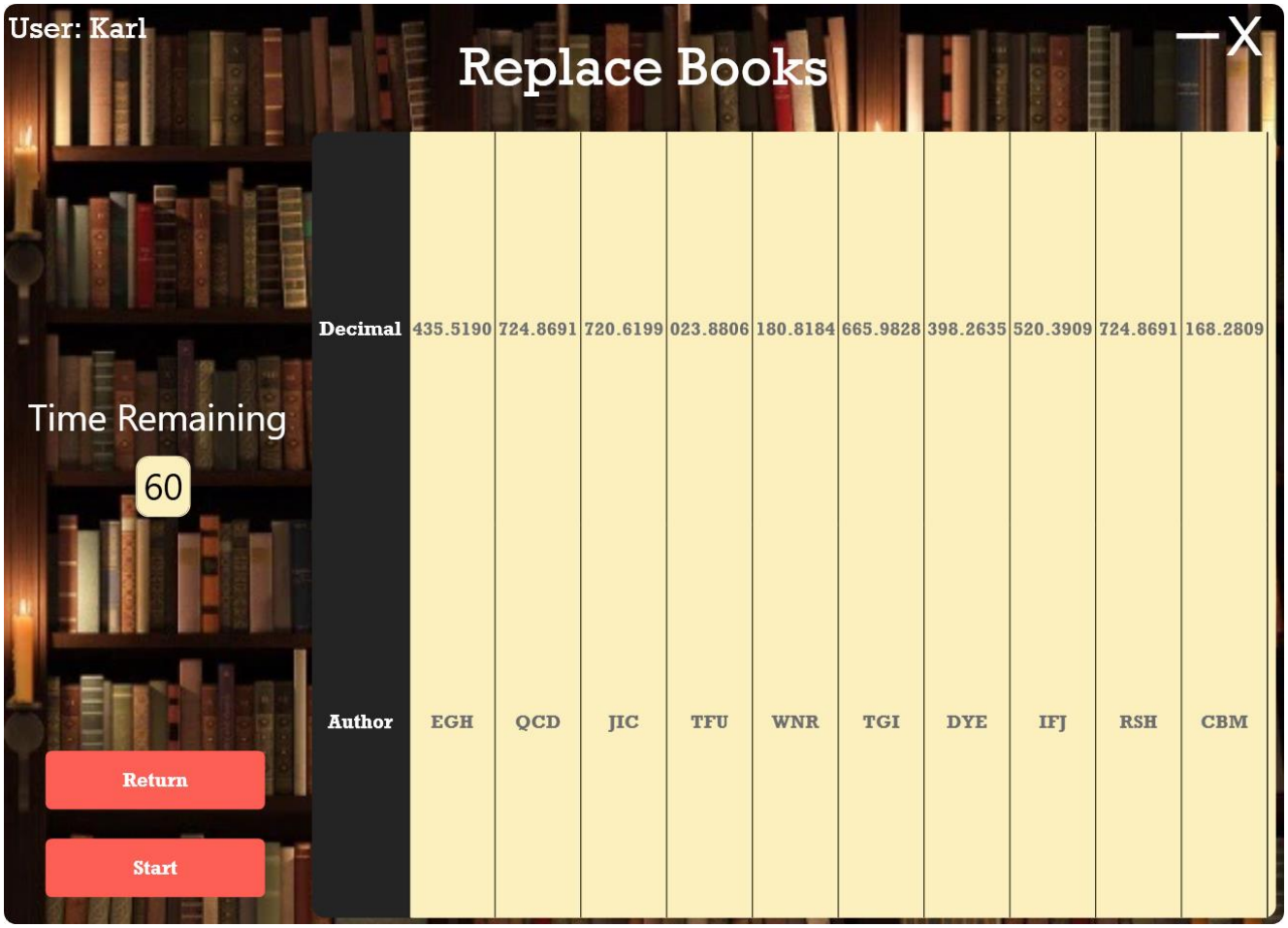


Once the user navigates to the register page, they can enter their account details, and press the “Register Account” button to create a new account on the system.

Once the user registers a new account, they are brought back to the login page, and can enter their account details.

The register page has input validation, so the passwords must match, and the username cannot be in use by another account.

Replace Books



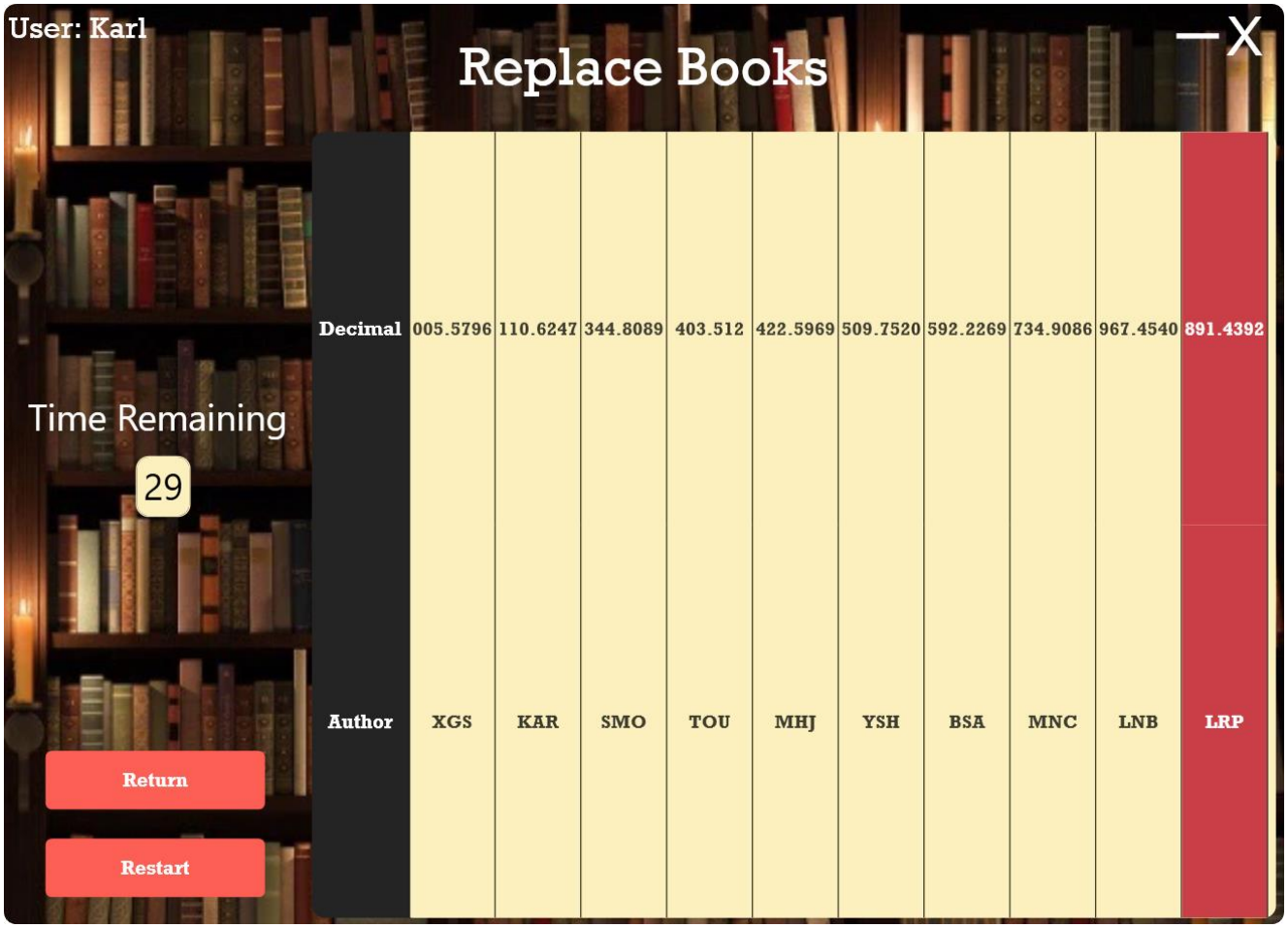
The “Replace Books” page can be accessed by pressing the “Replace Books” button on the main menu, which opens a new “game” or training session.

The replace books training session works by getting users to re-order the randomly generated call numbers, in numeric and alphabetic order – just like the Dewey Decimal system describes.

Once the user is ready to start the training session, they can press the “Start” button, which will refresh the call numbers and enable dragging of the Dewey decimals on the data grid.

Once the order is correct, the user will immediately be navigated to a confirmation page, where they can view the model answer, or return to the main menu.

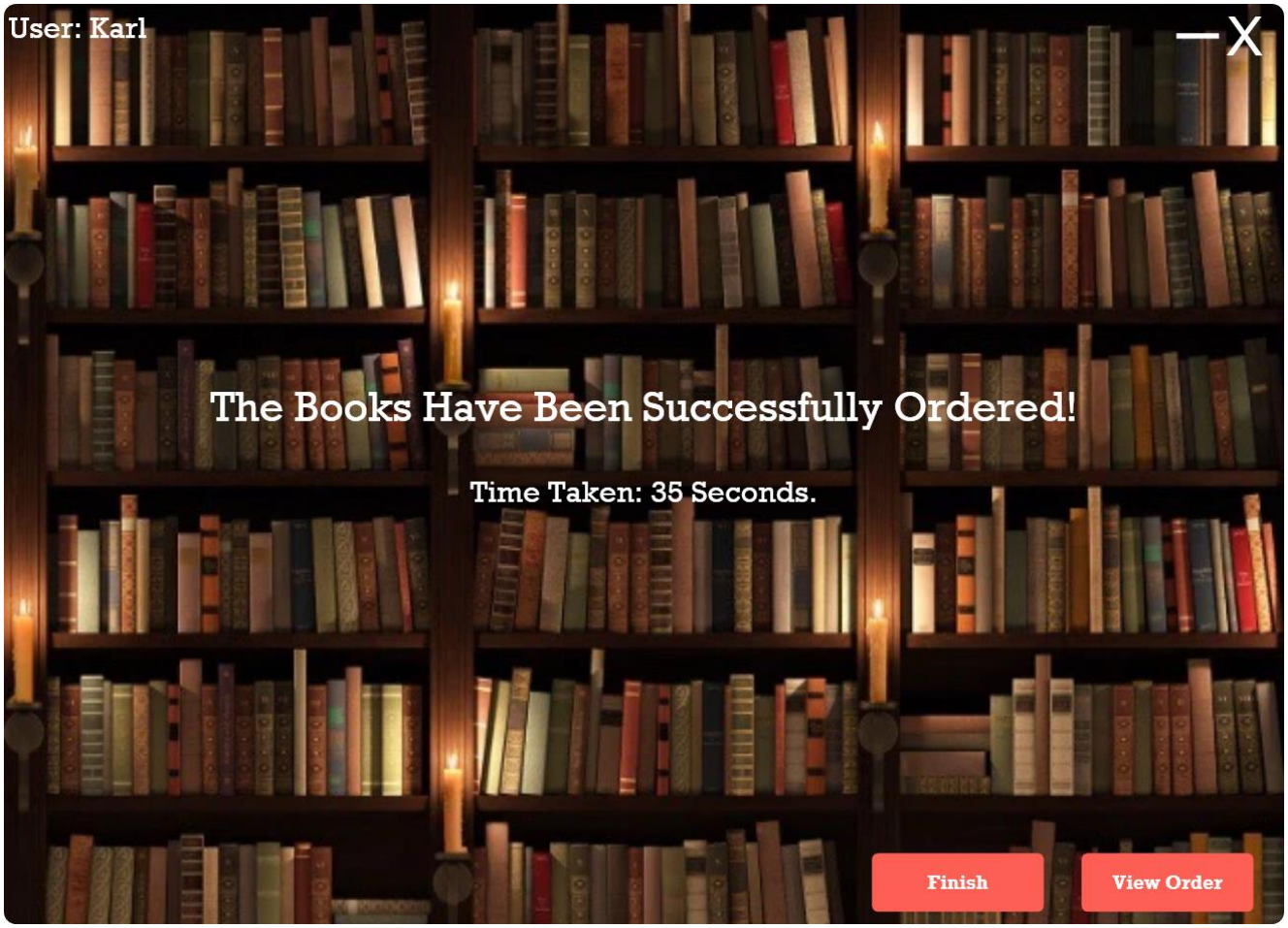
Replace Books



The user may re-order the books by dragging them across the page (clicking, holding, and moving them), which will allow the books to be re-ordered.

As can be seen on the provided image, the books have been partially ordered, and the timer is ticking down from 60 seconds, as the difficulty level has been set to “Easy” on the home page.

Confirmation Page



As soon as the correct order has been reached (once the call numbers are in their correct order), the user will be navigated to a confirmation page.

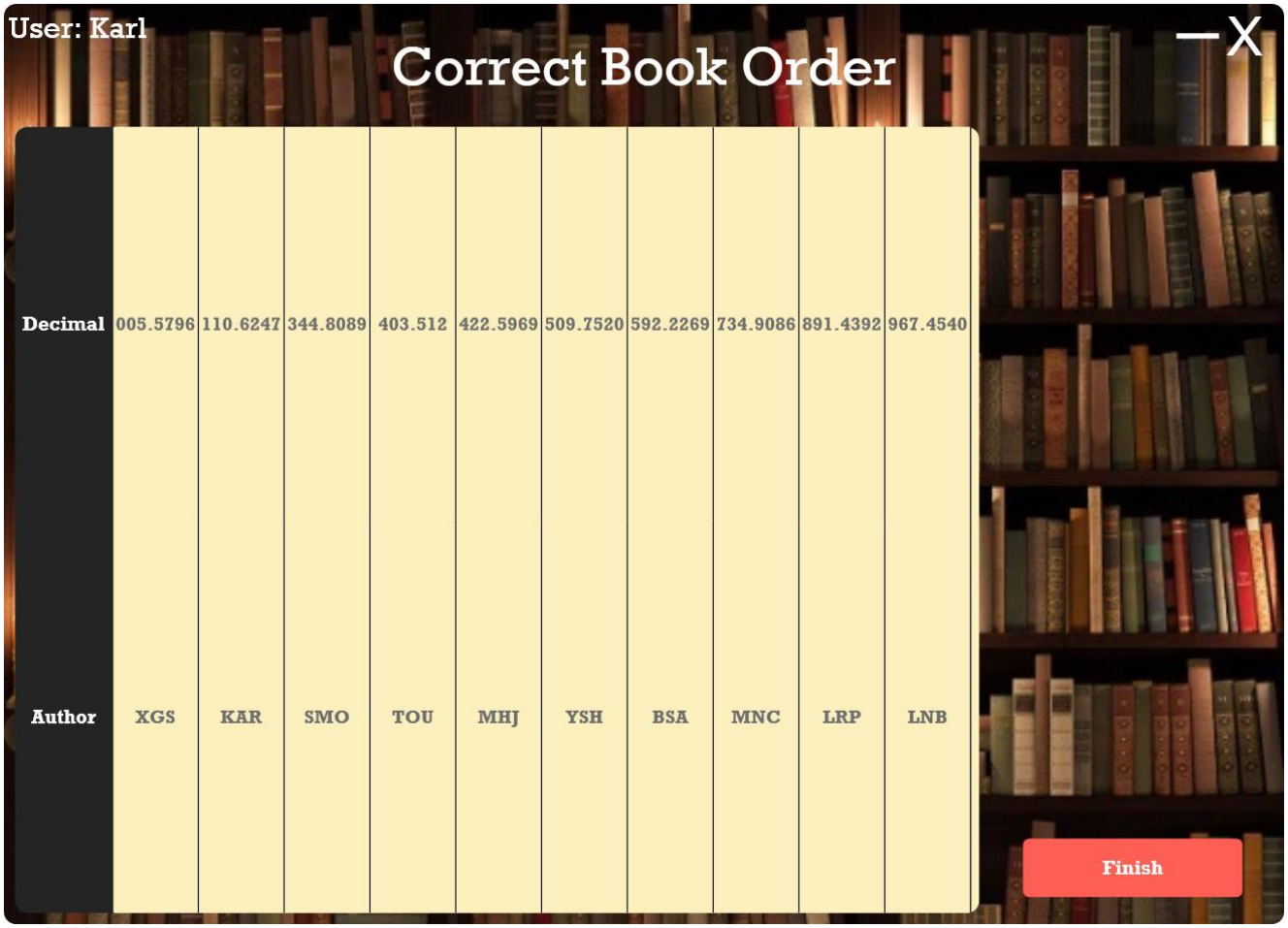
Firstly, this page will show whether the user has successfully ordered the books, with a confirmation message, and the time it took them to complete the session.

This confirmation page also provides the user with the ability to “View Order”, which allows them to access the model answer for the training session. Users can press the “View Order” to view this page.

It also allows them to return to the main menu by pressing the “Finish” button.

The “View Order” model answer page is shown on the following page.

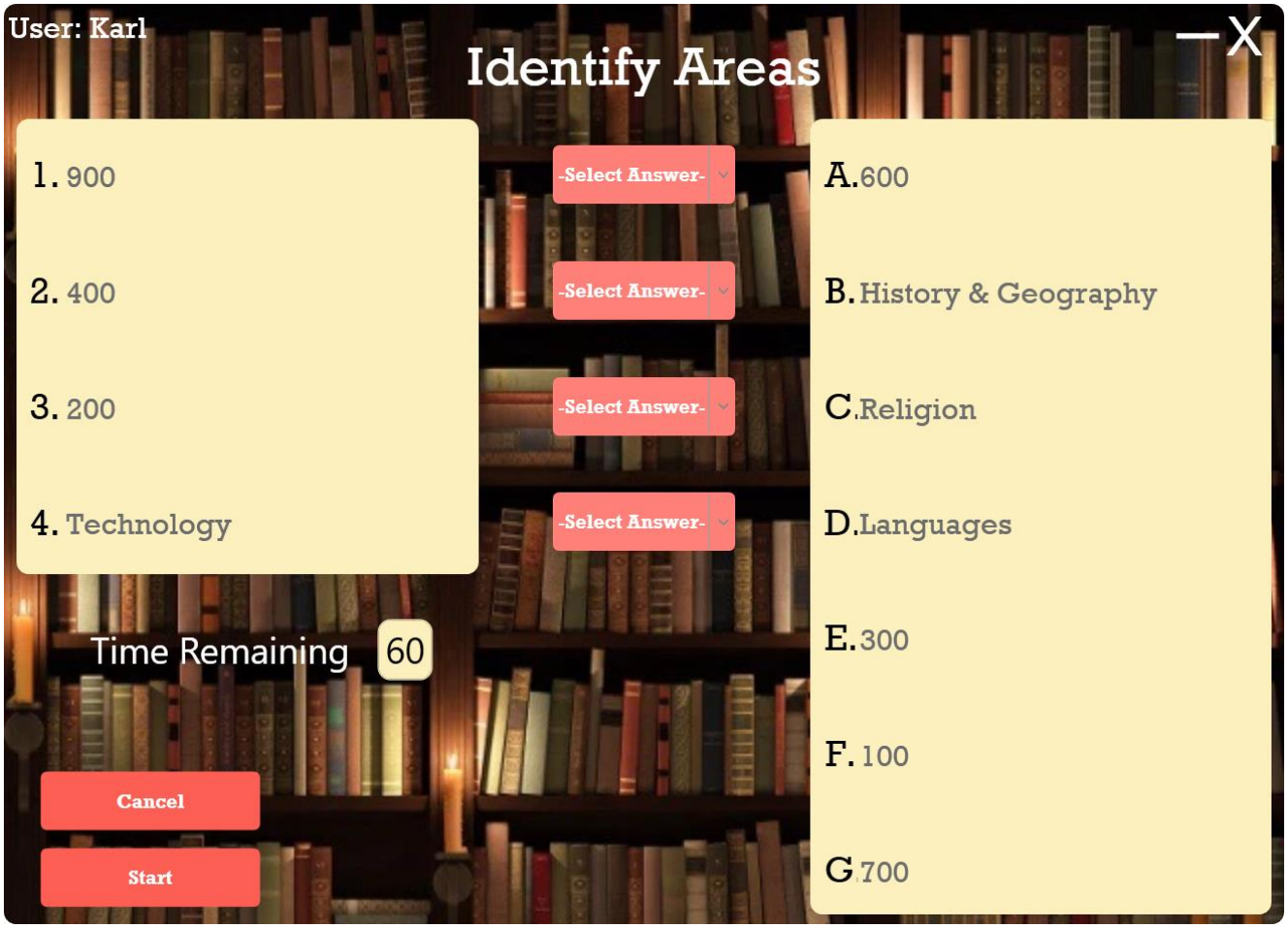
Correct Book Order



The “View Order” page is provided on the left, which shows the correct order of the Dewey decimal call numbers.

The user may press the “Finish” button to navigate back to the main menu after they have viewed the correct order for the call numbers.

Identify Areas

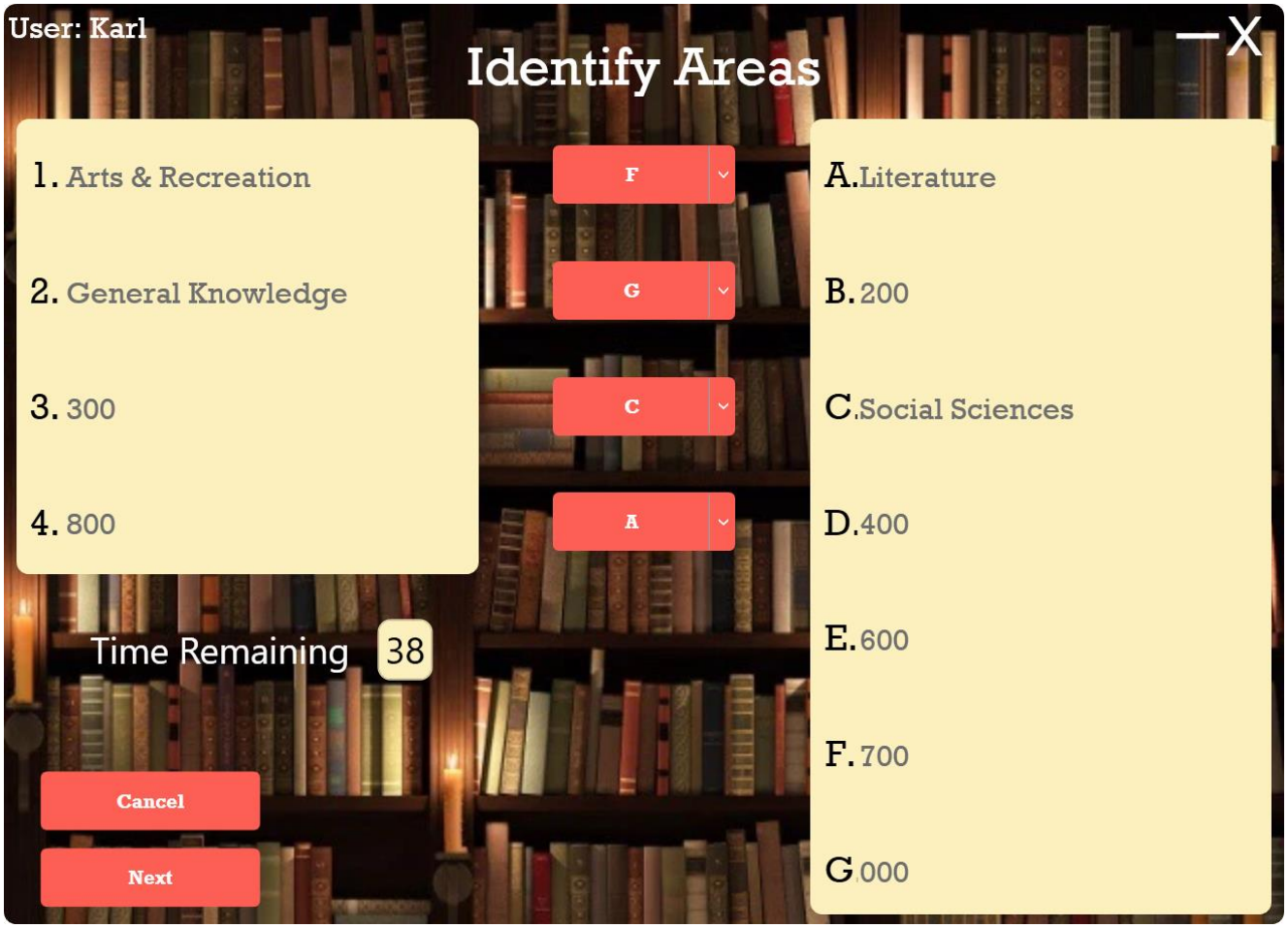


The identification of areas can be accessed by pressing the “Identify Areas” button on the main menu, which will navigate the user to the provided page.

This match-the-column training exercise provides the user with 4 randomly picked categories within the Dewey decimal system, and 7 potential answers on the right-hand side.

The user can press “Start”, which will randomize the questions and answers again, and will allow the user to select the correct answers from the dropdown boxes in the middle of the page – shown on the following page.

Identify Areas



The user may select the “Start” button, and select all their answers from the dropdown boxes next to each question.

For example, the image provided shows the answers to the provided training session.

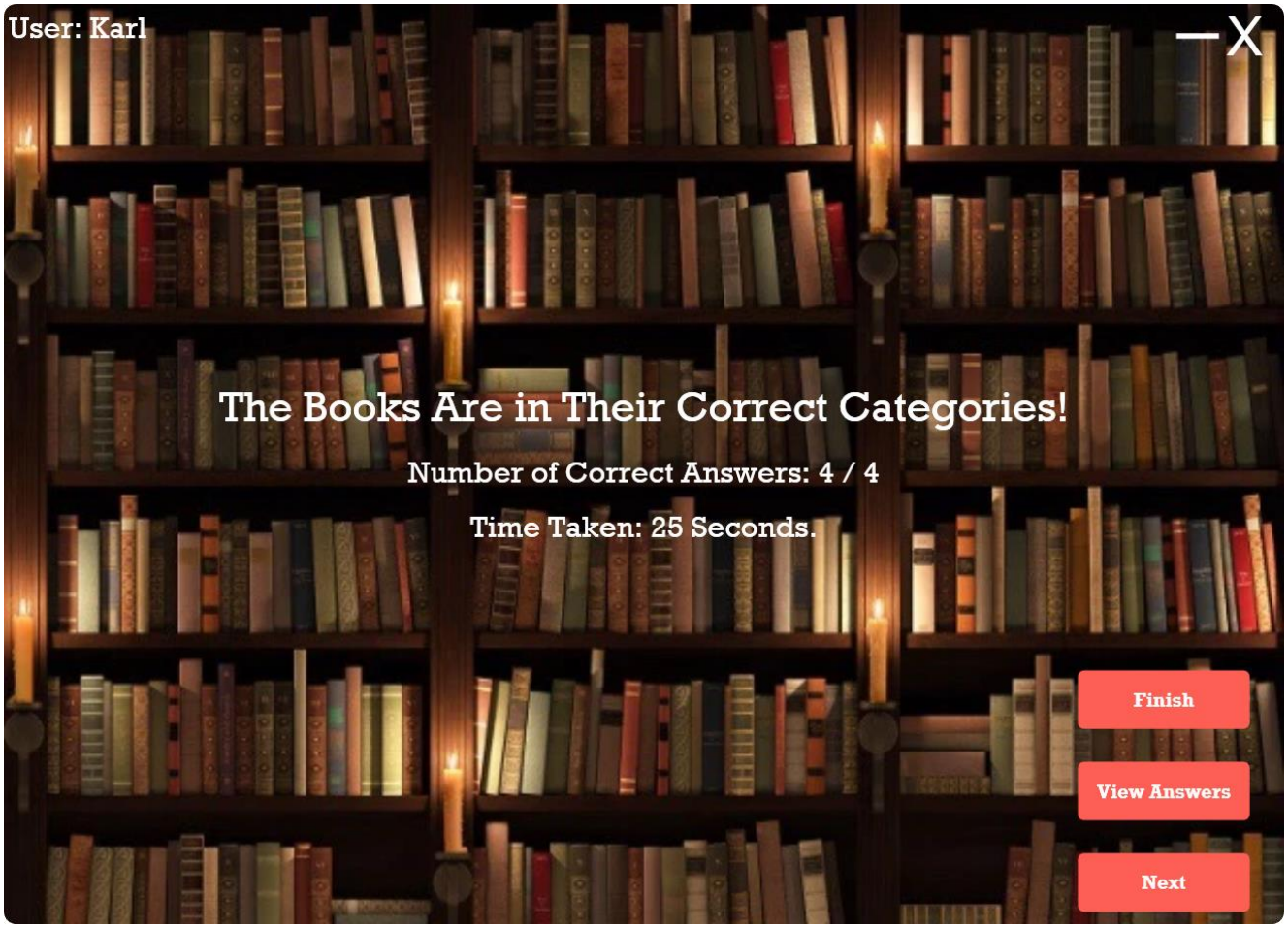
This training system also has gamification techniques implemented, in the form of a countdown timer, and logging of scores, much like the “Replace Books” exercise.

Once columns have been matched, by selecting an answer for each question from the dropdown boxes, the user can select “Next” to navigate the user to the confirmation page.

Input validation has been implemented on this page, so all inputs have to be valid.

Shown next is the confirmation page.

Confirmation Page



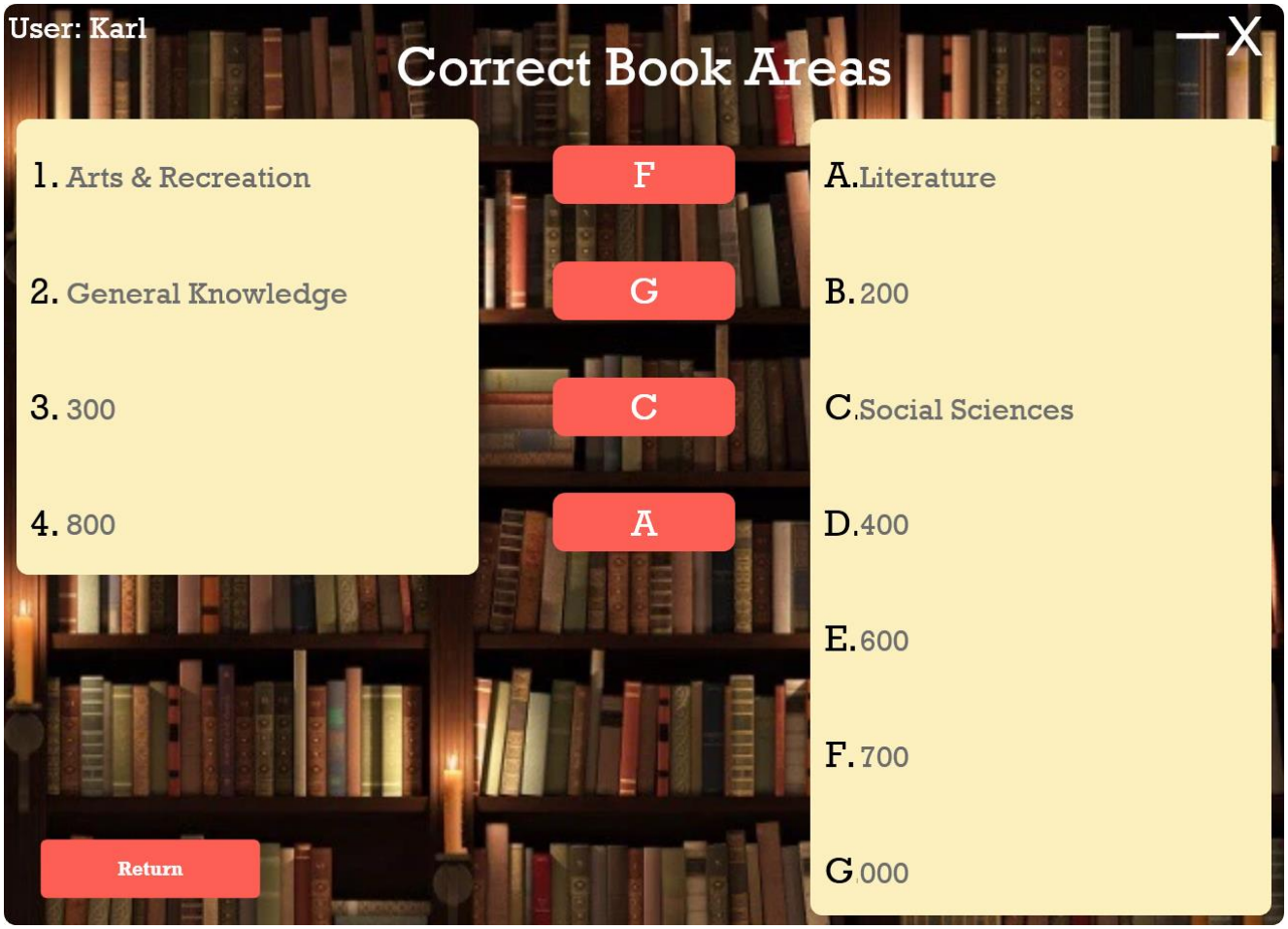
Once the user has selected “Next” or the timer has run out, the user will be navigated to a confirmation page, where it will be determined if all answers were correctly answered (the columns were matched correctly).

The user will be presented with the number of correct answers, and time taken to complete the test.

The user can navigate back to the main menu by pressing the “Finish” button, view the model answer by pressing “View Answers”, or press “Next” for another “Identifying Areas” training session.

The model answer page is provided on the following page, where the answers for all questions are provided.

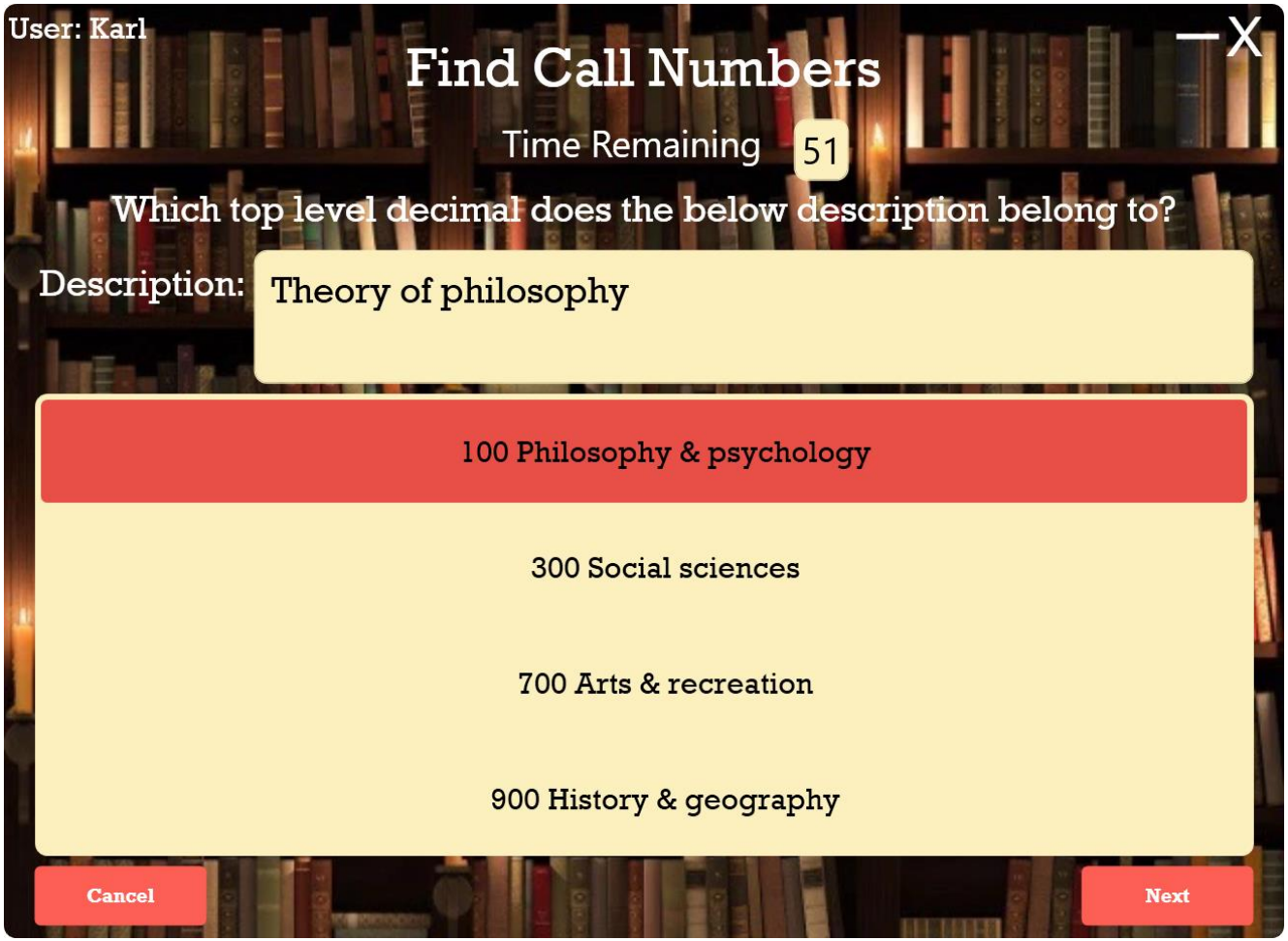
Correct Book Areas



If the user presses the “View Answers” button, a page with the model answer will be provided, so that the students and librarians can learn from the system, and not only test their knowledge of the Dewey decimal system.

If the user presses the “Return” button, they will be brought back to the confirmation page, where they can continue with another identifying areas session, or finish the game.

Find Call Numbers – Level One

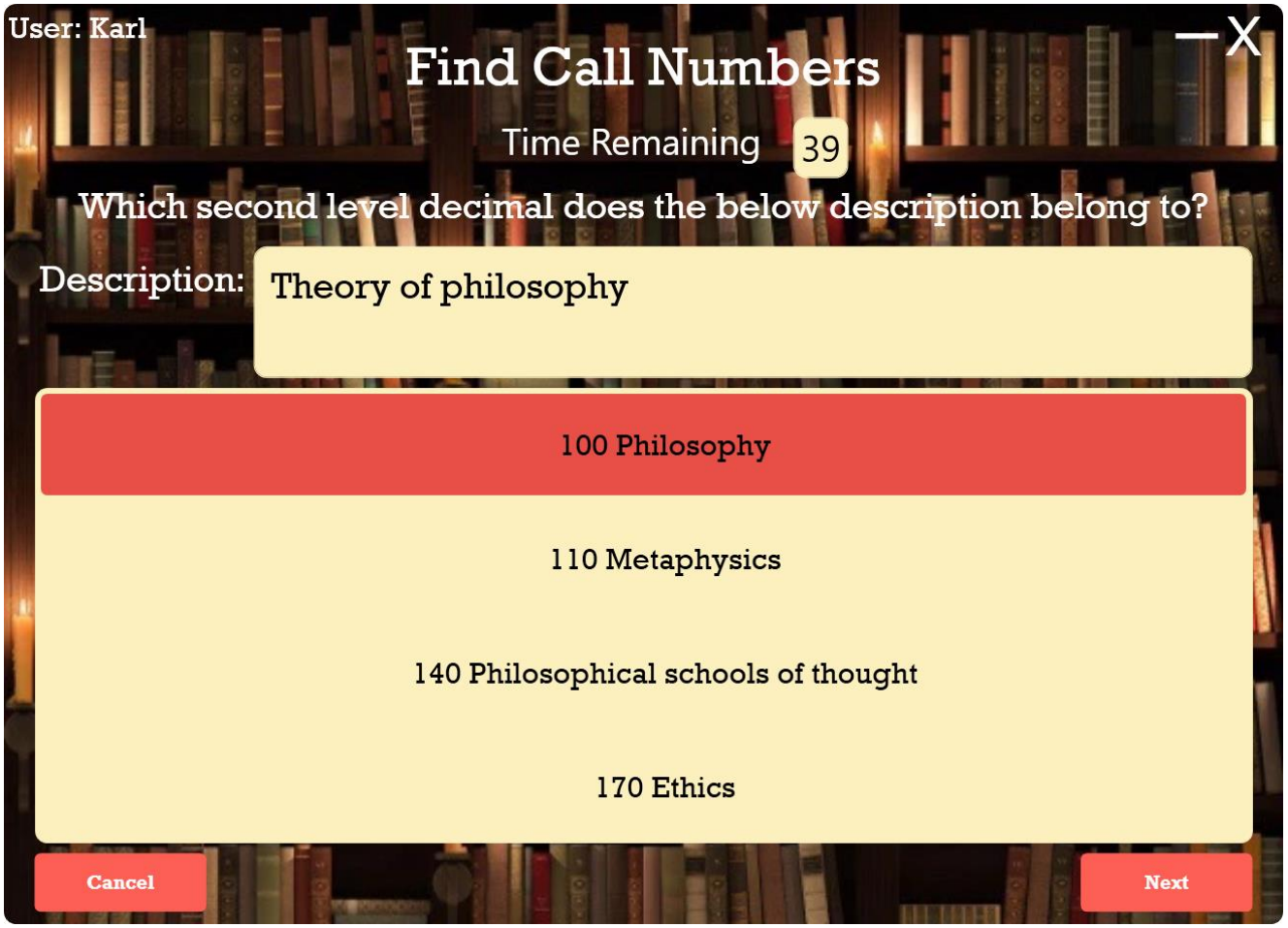


The finding call numbers feature can be accessed by pressing the “Find Call Numbers” button on the main menu, which will navigate the user to the provided page.

A tree data structure allows the user to drill down from level one call number / description to the third level call number / description. The quiz allows users to view a third level description and first level, second level, and then third level call numbers / descriptions. Three of which will be incorrect, with one being correct.

The user can press “Next”, which will bring the user to the next level, and so forth until the third level has been reached. At any stage if the question is incorrectly answered, then they will be navigated to a page where a message will be displayed (incorrect selection), and they can start a new test from this page or view the model answer.

Find Call Numbers – Level Two

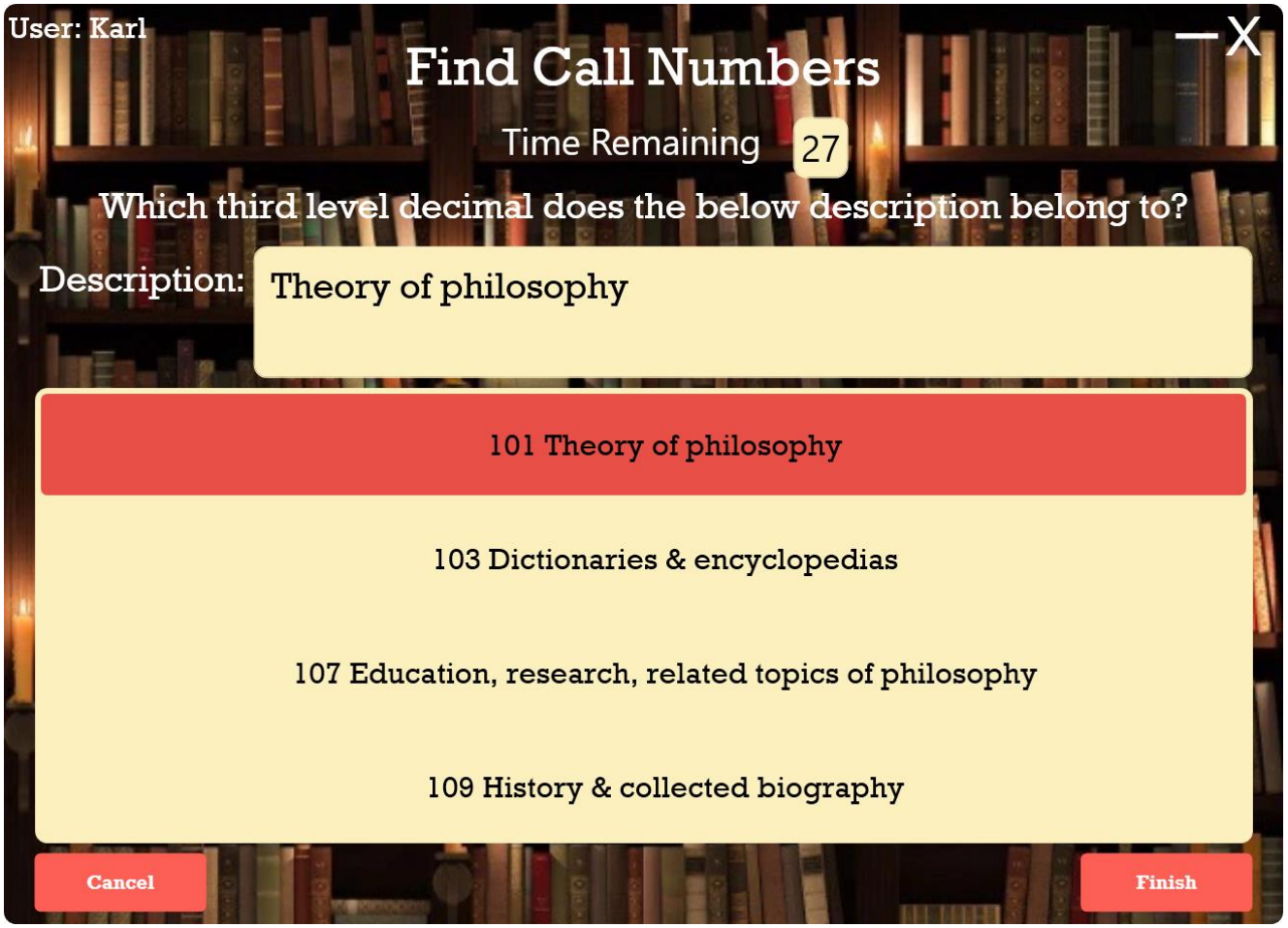


Once the user has selected the correct first level Dewey Decimal / description, they can then view the second level answer and press “Next”, until they reach the third level decimal / description.

If at any point they select an incorrect first / second / third level decimal / description combination, they will be navigated to a confirmation page, where a suitable message will be displayed for the incorrect selection.

The timer will tick down until the user has selected an entry and clicked the “Next” or “Finish” button.

Find Call Numbers – Level Three



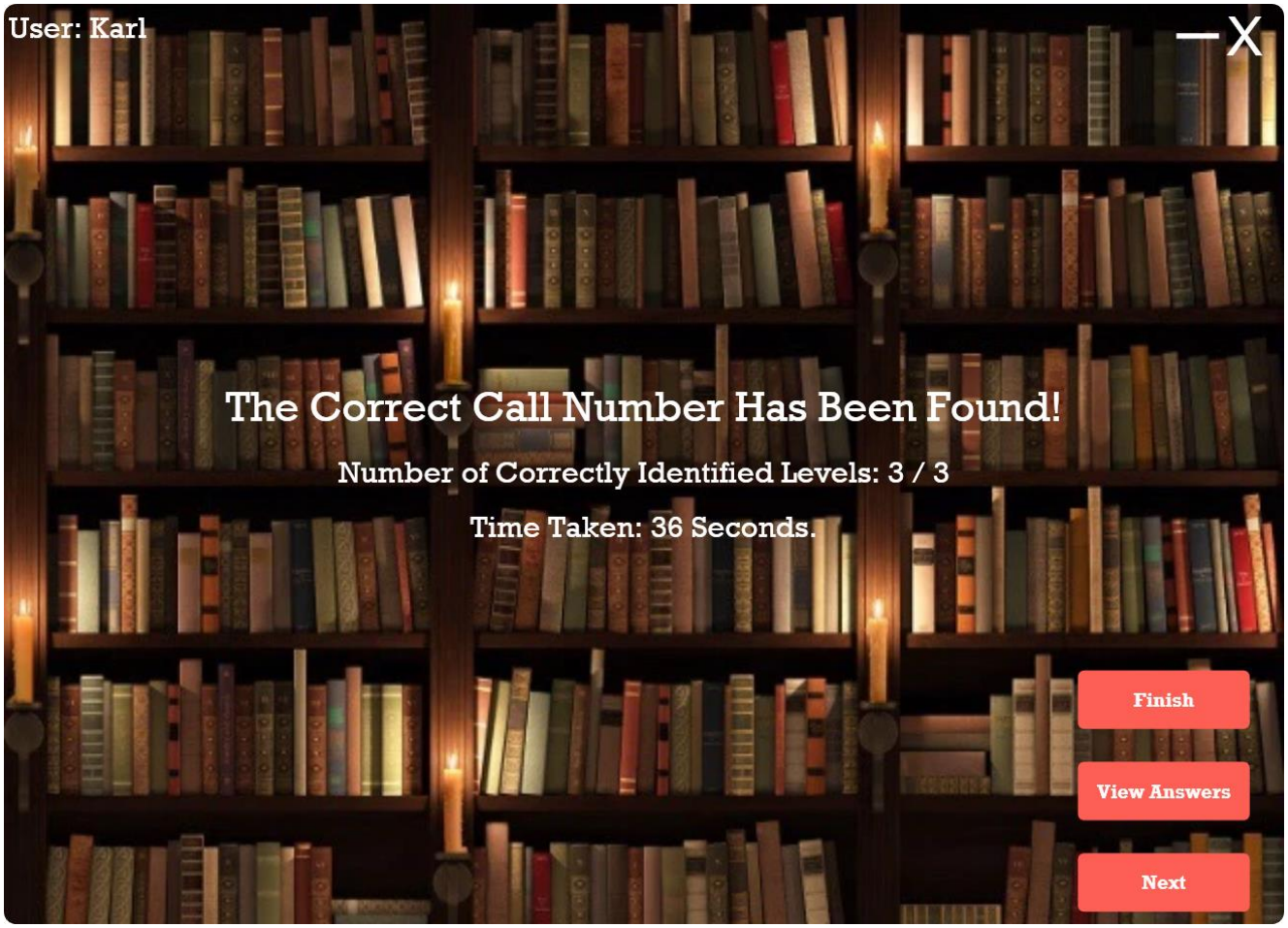
Once the user has selected the correct second level Dewey Decimal / description, they can then view the third level answer and press “Finish” to complete the test.

If at any point they select an incorrect first / second / third level decimal / description combination, they will be navigated to a confirmation page, where a suitable message will be displayed for the incorrect selection.

The timer will tick down until the user has selected an entry and clicked the “Finish” button.

Once finished, the user can view the model answer, navigate to the home page, or continue to another question, which will start a new finding call numbers test.

Confirmation Page



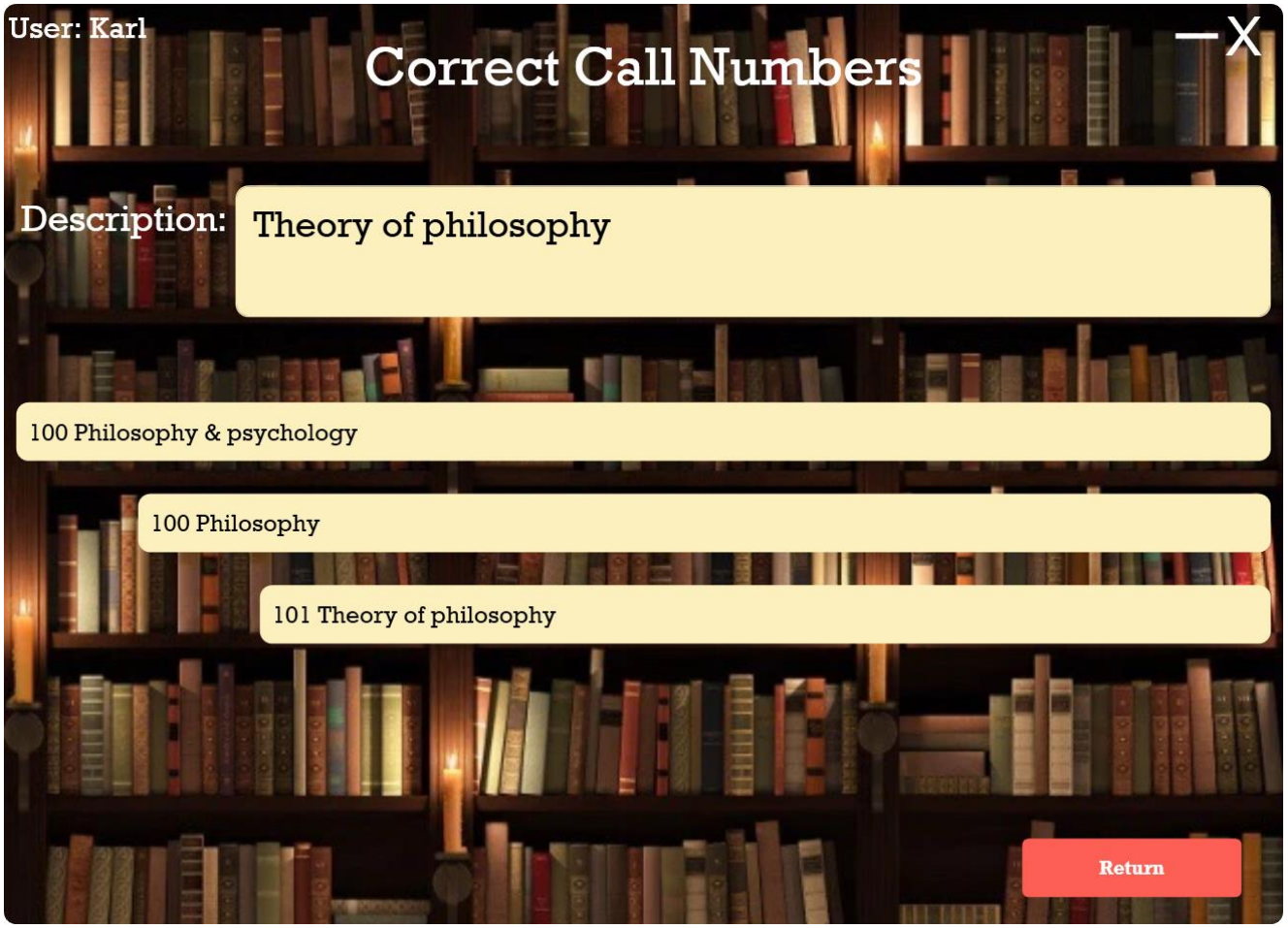
Once the user has selected “Finish”, the user will be navigated to a confirmation page, where it will be determined if all answers were correctly answered (the levels were correctly selected).

The user will be presented with the number of correct answers, and time taken to complete the test.

The user can navigate back to the main menu by pressing the “Finish” button, view the model answer by pressing “View Answers”, or press “Next” for another “Finding Call Numbers” training session.

The model answer page is provided on the following page, where the answers for all questions (descriptions) are provided.

Correctly Found Call Number



If the user presses the “View Answers” button, a page with the model answer will be provided, so that the students and librarians can learn from the system, and not only test their knowledge of the Dewey decimal system.

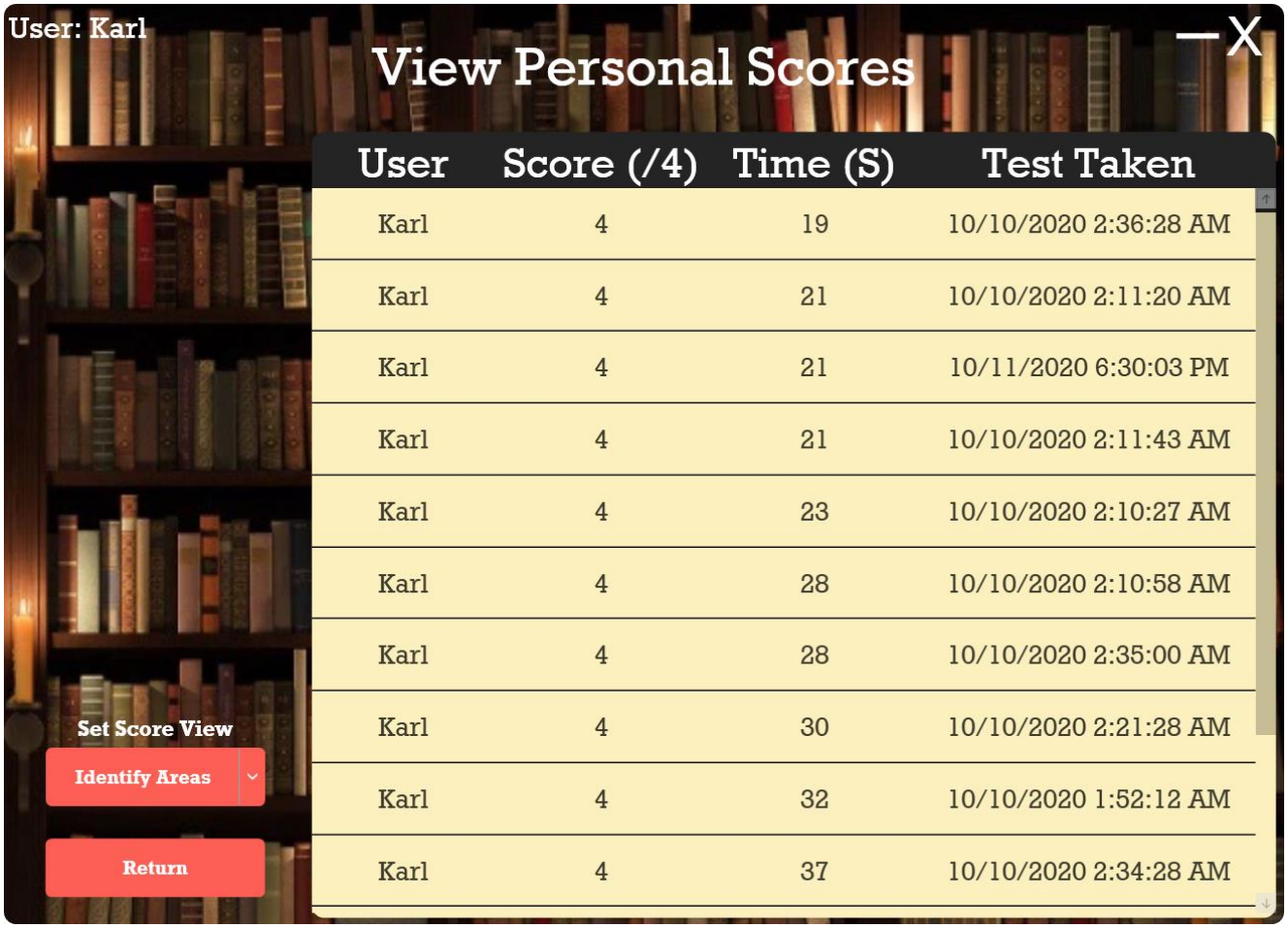
If the user presses the “Return” button, they will be brought back to the confirmation page, where they can continue with another finding call numbers session, or finish the game.

View All Scores – Replacing Books

If the user presses the “View All Scores”, a page with all their personal scores will be displayed for both the “Replace Books”, and “Identify Areas” training sessions.

If the user wishes to view scores for the “Identifying Areas” sessions, they can select that option from the dropdown just above the “Return” button, and all their scores for that game / training type will be displayed.

View All Scores – Identifying Areas



If the user selects the dropdown box from above the “Return” button, they are presented with all “game” types, including “Replace Books”, “Identify Areas”, and “Find Call Numbers”.

This can be set on the main menu page as well, where the top ten scores are shown.

The following image shows the scores for “Identify Areas” training sessions for the logged in user.

This includes the username, score, time taken to complete the test, and the time that the test was taken.

View All Scores – Finding Call Numbers

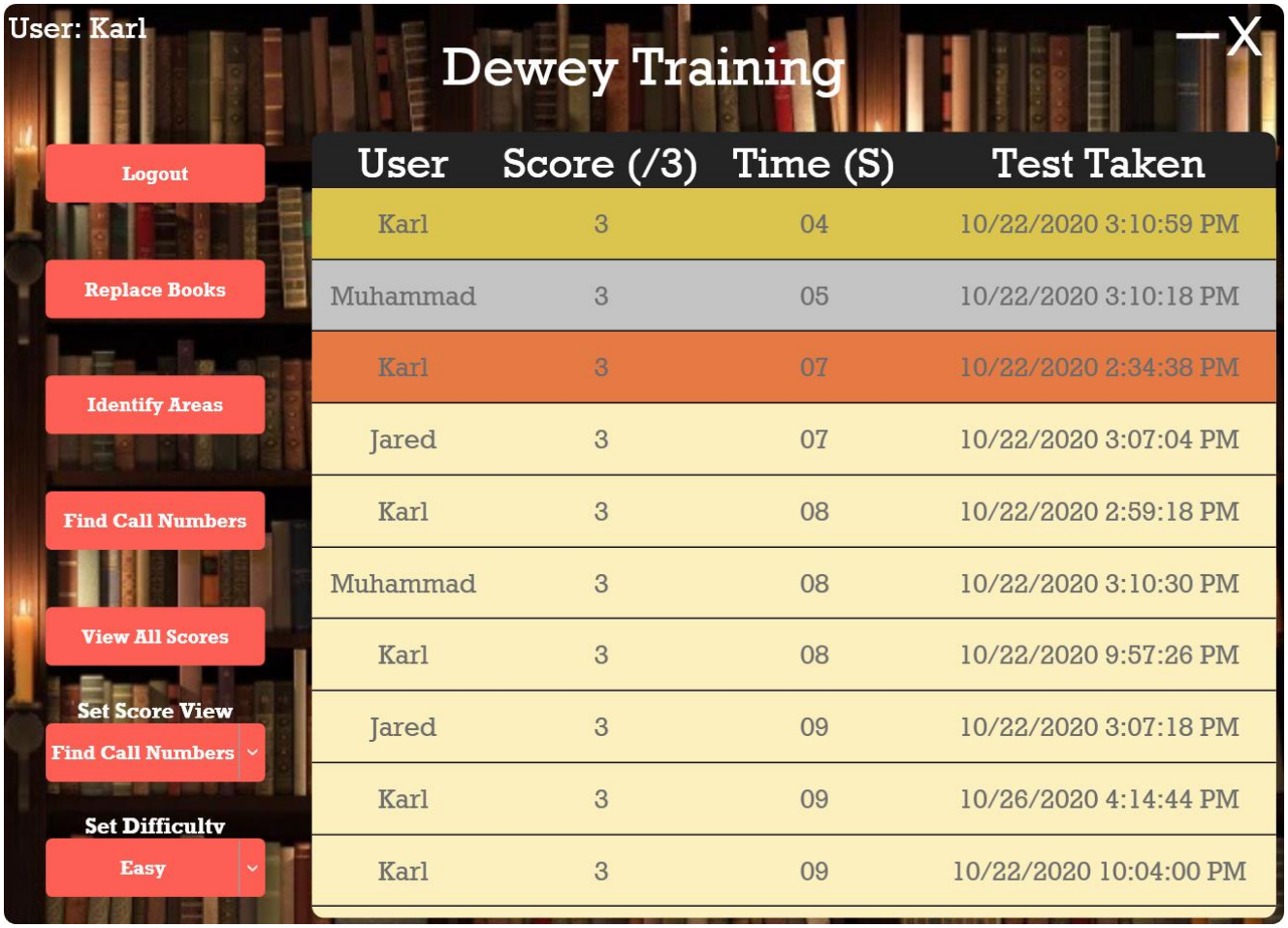
If the user selects the dropdown box from above the “Return” button, they are presented with all “game” types, including “Replace Books”, “Identify Areas”, and “Find Call Numbers”.

This can be set on the main menu page as well, where the top ten scores are shown.

The following image shows the scores for “Find Call Numbers” training sessions for the logged in user.

This includes the username, score, time taken to complete the test, and the time that the test was taken.

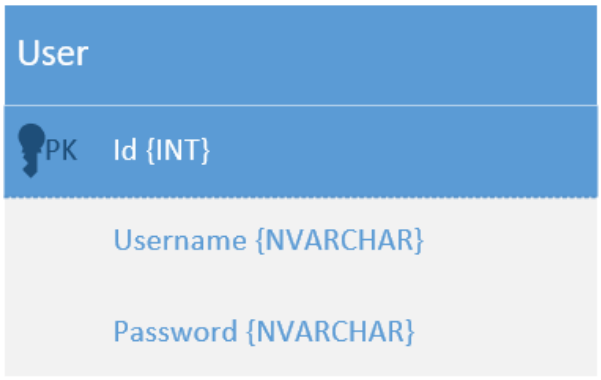
Set Game Difficulty



If the user wishes to change the game difficulty, by reducing the total time that is allowed for each training session, they can set the difficulty – the lowest dropdown element on the provided screenshot.

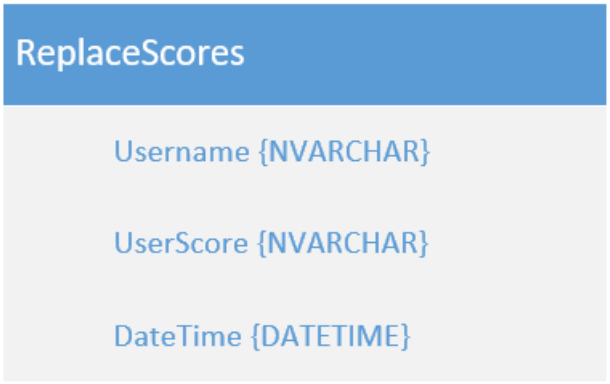
This will set the times of the counter to 60 seconds for Easy, 40 seconds for Medium, and 30 seconds for Hard difficulties.

Database Entities

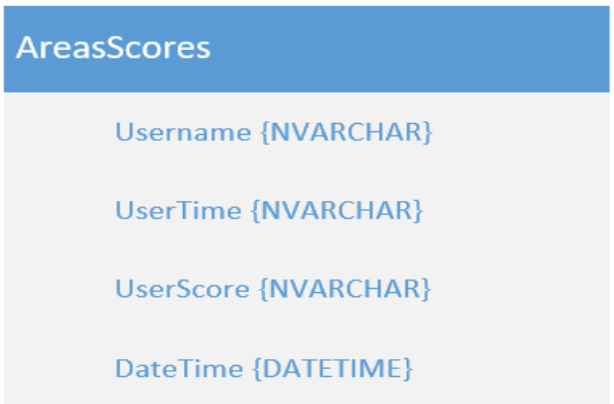


The User model defines what is saved for each user in the system. This includes the user id, username, and password.

The Categories model defines what information is stored for each Dewey Decimal category (for Task 2).



The ReplaceScores model defines what is saved for each replace books test score entry in the database, which includes the username of the user who achieved the score, the score value, and the date and time that the entry was inserted into the database.



The AreasScores model defines what is saved for each identifying areas test score entry in the database, which includes the username of the user who achieved the score, the time it took to complete the test, the score value, and the date and time that the entry was inserted into the database.

Tree Data Structure

The tree data is stored in a CSV file within the Dewey Training\TreeData directory. This data is pulled into a tree structure within the application, and each delimiter (!, #, etc) denotes a particular action to take – e.g. “!” creates a new node, “#” creates a new child node, and so forth.

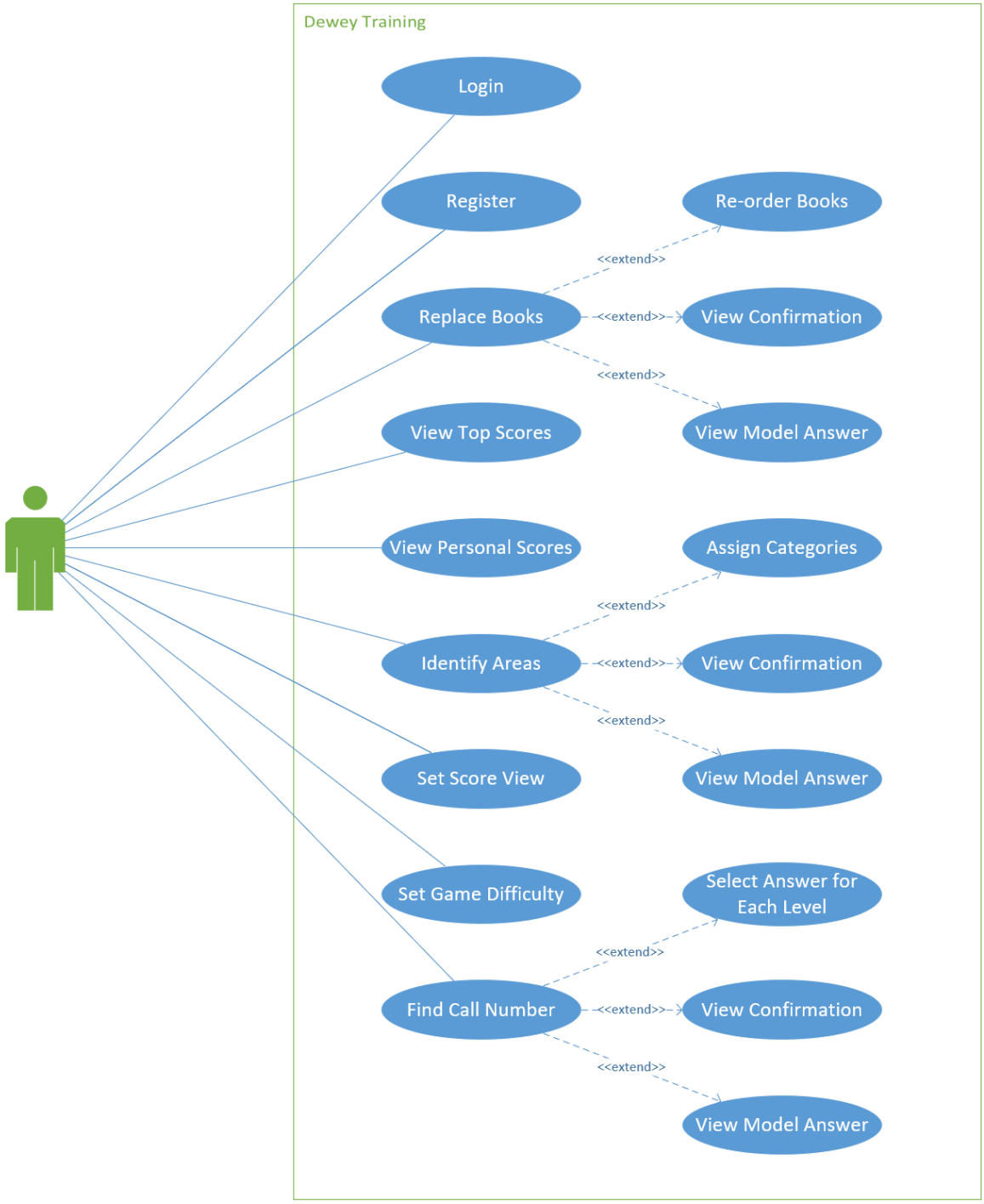
An example of the contents is provided below:

```
!-000$^Computer science, information & general works
@--000$^Computer science, knowledge & systems
#---001$^Knowledge
#---003$^Systems
#---004$^Data processing & computer science
#---005$^Computer programming, programs & data
<>
@--010$^Bibliographies
#---012$^Bibliographies of individuals
#---015$^Bibliographies of works from specific places
#---017$^General subject catalogs
#---019$^Dictionary catalogs
<>
@--020$^Library & information sciences
#---023$^Personnel management
#---025$^Library operations
#---026$^Libraries for specific subjects
#---028$^Reading & use of other information media
<>
@--030$^Encyclopedias & books of facts
#---031$^Encyclopedias in American English
#---032$^Encyclopedias in English
#---034$^Encyclopedias in French, Occitan, and Catalan
#---035$^Encyclopedias in Italian, Romanian, and related languages
<>
<>
```

This data is pulled into a tree structure in the application and used to create a multi-level question system.

Each delimiter splits the string values in a very specific way as to create this multi-level list in the system, allowing for comparisons.

Use Case Diagram



Conclusion

In conclusion, this documentation has provided extensive development information in order to detail how and why the Dewey Decimal desktop application was developed in the way that it was. It described each function of each page within the “Help File” section, and provided user interface design information within the “Screenshots” section. A “Readme” section was also included in the document to provide information about the development environment, instructions on how the desktop application should be run, and other such critical information to get the application running on the user’s PC.
Additional information such as all the database entities was provided, in addition to the tree data file, which detailed how and where data was stored by the application.
In addition to the above, a use case diagram was included, which showed core functionality of the desktop application from the user’s perspective.

During the course of this project, we have learnt how to develop advanced C# desktop applications in the .Net Core 3.1 Framework. We also learnt how to use a Data Access Layer – DAL to access information from a local MDF file, and later on this will be hosted online. In addition to this, we have learnt how to use advanced data structures, including Doubly Linked lists, Dictionaries, Key Value lists, Observable Collections, and Tree Structures, in addition to other datatypes.

This is the final POE project, and therefore feedback has been incorporated into the applications, and research, which can be found in their respective folders in this submission.

References

Jaboss. (n.d.). *Library Icon*. Retrieved from Cleanpng: https://www.cleanpng.com/png-library-computer-icons-librarian-reference-work-cl-5637031/

Laja, P. (n.d.). *How to Use Gamification for Better Business Results*. Retrieved from Neilpatel: https://neilpatel.com/blog/gamification-for-better-results/

Quicksprout. (2016, 07 15). *How to Easily Add Gamification Techniques to Your Content*. Retrieved from Quicksprout: https://www.quicksprout.com/how-to-easily-add-gamification-techniques-to-your-content/

The Independent Institute of Education. (2020). *PROG7312 POE Question Paper.* Sandton: The Independent Institute of Education.

Wolfie. (2019, 01 28). *Library Background Image by Wolfie*. Retrieved from Picsart: https://picsart.com/i/286394316039201