COS60010 - Technology Inquiry Project

Semester 1 - 2024

Deliverable 3

Project Delivery Documentation

Group Name: GROUP 1

Student members: Arun Ragavendhar Arunachalam Palaniyappan 104837257

Christopher Opie 105380202

Layan Buddhimal Weerasingha Arachchige 104758921

Amirajsinh Pradhyumansinh Sonagara 104801333

Henil Mukeshbhai Pistolwala 105065800

Workshop: Thursday – 14:30-16:30 - Room EN207

Facilitator: Dr. Alfandi Yahya

Contents

1. Introduction

The main goal of this document is to discuss in detail the following 3 topics:

Project Specification Documentation for Business Usage:

This section provides an overview of the project definition, project objective, the intended target users of the project as well as the features implemented in the project from a business and usability point of view.

Software Requirements and Design Specifications for Technology Usage:

This section provides a detailed explanation of the features implemented from a programming point of view with the technical details, code base, design architecture and the database schema specifications.

User Manual:

This section provides a step by step guiding on how to navigate, use and benefit from the application.

The objective of this document is to discuss in depth what has been built, why it has been built, and how it has been built, from a business point of view as well as a technical programming point of view.

2. Project Specification Documentation for Business Usage

2.1 Project Definition:

Instatute Pvt.Ltd is a well-known tutoring company present at various locations throughout Australia. Currently, they offer one-on-one and group tutoring services to help high schools students prepare for university entrance exams for both Australian as well as international institutions.

Instatute is currently exploring gamification to enhance and improve engagement and participation in their teaching methods. Their internal company policies and legal constraints restrict the use of use software tools to those that have been specifically built for them.

Hence, after conducting a detailed technical business analysis and user requirements study with the client Instatute Learning Pty Ltd., this interactive Chem Quiz application has been designed and developed to support the preparation of their students for university entrance exams for both Australian and international institutions. The Chemistry quiz application is web based and has been developed to make the learning easier and more engaging for students struggling with challenging chemistry concepts.

2.2 Project Objectives:

The quiz has been implemented as a support application which is available within the main website of Instatute Learning Pty Ltd. The core idea is that every enrolled student at Instatute

Learning Pty Ltd will have access to this application. A student user can login using his/her credentials and can play the game an unlimited number of times. They can modify the difficulty level and view their historical results, as well as the results of their peers through a leaderboard, to effectively evaluate where they stand.

The application also provides an admin login for a staff, allowing them to create and modify questions, and provides them with the ability to modify student records.

2.3 Intended Users

The target audience of this application are high school students who are currently studying chemistry. The app at its core is an interactive chemistry quiz where a student can view a set of chemistry related questions and answer them. Once finished, they can then instantly see their score, providing instantaneous feedback. The currently implemented question types are both structure and reaction based, meaning that the student would be tasked to identify structures, determine the products of chemical reactions, and draw chemical structures to answer the questions. This helps to enhance the analytical and cognitive abilities of the student and helps in solidify their knowledge.

2.4 Research Report

Once the user requirements were obtained from Instatute, the group members set out to perform extensive research to aid them in the following activities:

- Convert the business requirements into user stories.
- Analyse the user stories and create a system design that can be implemented into a working project.
- With the system design is locked in, start researching technology stacks to program and develop the project.
- After a thorough research and reflection, the application has been designed and developed using HTML, CSS, JavaScript, PHP and MSQL. Further details of which has already been described in detail in the project concept report.
- The references used have been duly mentioned and credited in the document.

3. User Stories, Implemented Functionalities and Priorities

Based on the requirements purpose discussed in the Introduction, the following user stories have been designed and their respective priorities designated.

3.1 Main functionality of the quiz (displaying questions and collecting answer inputs) for chemical structure-based and reaction-based chemistry questions

Priority: 1 (Highest)

- Students are given a series of chemistry-based quiz questions displayed to them, and they can submit their answers to the application.
- The focus is on chemical structure-related and reactions related questions because it is fundamental to many branches of chemistry and is a challenging topic for students.

- This implementation helps to distinguish the program from existing chemistry-based quizzes.

3.2 Displaying a list of correct and incorrect answers at the end of the quiz

Priority: 2

- For an effective self-assessment, players are given feedback at the end of a game which displays the questions asked, their responses, and the correct answers.
- Students are then encouraged to revisit their textbooks or notes, to clear up any misunderstandings they may have.

3.3 Incorporating username/password-based login & authentication for students

Priority: 2

- Students can access the “Chem Quiz” lobby by entering their username and password which has been provided to them by the Instatute Learning Pty Ltd.

3.4 Recording and retrieving player statistics and scores via the database

Priority: 2

- Student scores are saved into the database after they attempt a quiz.
- Statistics of their previous attempts are provided to them after they log into the application, such as their highest score, scores of the last five attempts, as well as a Leader Board where they can see peer performances and evaluate where they stand.
- At the end of each attempt their score is immediately updated with the latest attempt, and the updated statistics displayed when they return back to the welcome page.

3.5 Project deployment to the web

Priority: 3

- The application has been deployed to the web as soon as Priority 1 and Priority 2 features have been implemented.
- Since the development environment is different to the production environment, an early deployment has provided ample time for testing and debugging.
- The project has been debugged and thoroughly tested and is running the latest stable version.

3.6 Admin access provided to teachers for database modifications

Priority: 4

- Teachers and staff have access to an administrative area of the application.
- They have access to perform tasks such as the addition, deletion or updating of quiz questions and student scores information.
- The addition and creation of questions feature specifically would allow teachers to provide a more customised experience for their students, resulting in questions being directly relevant to the students' current studies.

3.7 Option to Choose Difficulty levels for questions

Priority: 4

- Students can modify the difficulty level of the quiz questions, since this application is meant to be used by at least high school students from a range of year levels.

3.8 Extension of the application with additional subjects and questions for the future

Priority: 5 (Lowest)

- Apart from serving it current usage, the quiz application has been designed in such a way that the source code can be worked upon in the future to further add new questions types as well as more subjects apart from chemistry. The scope of this depends upon the client requirements and the response and feedback from the students, but, when the need arises, the application is ready to be upgraded.

3.9 Future scope and plans for Application Scalability

Priority: 5 (Lowest)

- The application is currently deployed on a remote server with a public IP address and is accessible to anyone who has access to the internet. Based on the student response and the client demands, the application can be containerized with its code, package dependencies and run time, and can be horizontally scaled and deployed on a set of distributed systems, if the need arises for a larger demand of the application.

4. Application Functionality and Usage after development

4.1 Login Page:

- Username/password-based login & authentication for Students and Admins
- Students /Admins can access the “Chem Quiz” application by entering their username and password which has been provided to them by the Instatute Learning Pty Ltd.
- Routes the user to the welcome page if they are a student.
- Routes the user to the admin page if they are an admin.

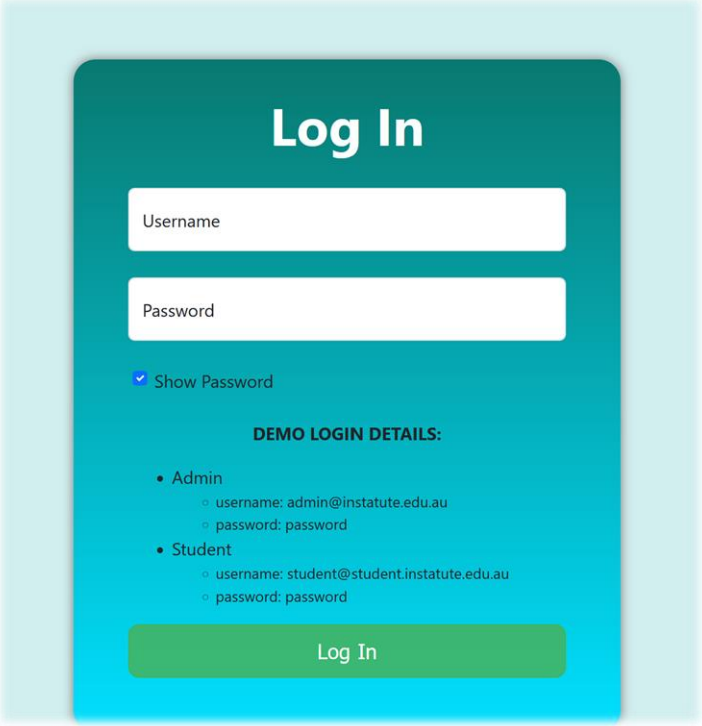


Fig. 1: User Interface of Login Page

4.2 Welcome Page:

- The logged in student can view statistics in the form of the scores of their previous attempts, a leaderboard, and the number of quiz attempts they have made.
- They can select a difficulty level for a single quiz attempt, before starting the game.
- They can also press the LOGOUT button and return back to the login page.

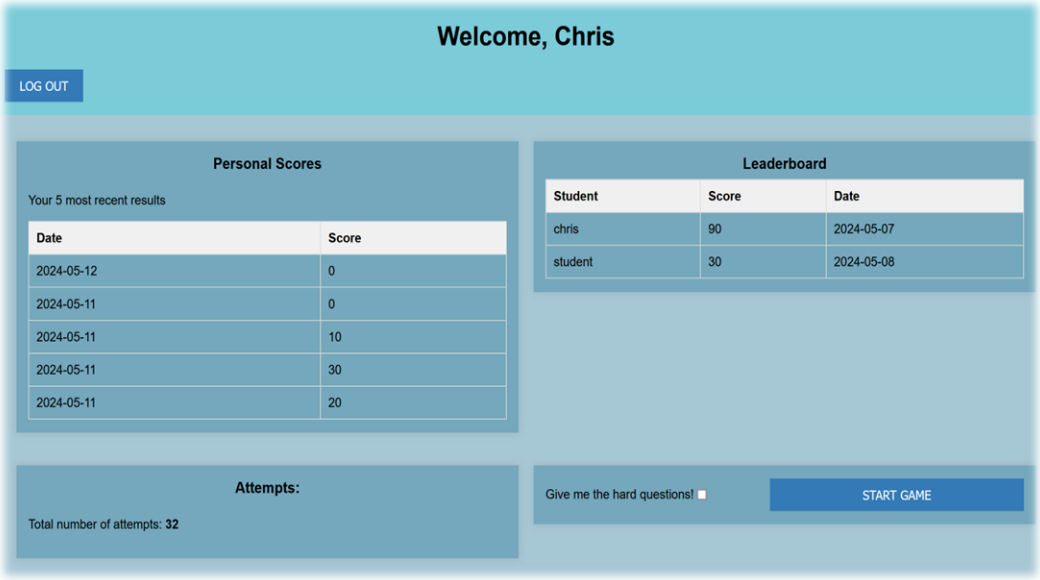


Fig. 2: User Interface of Welcome Page

4.3 Questions Page:

- Displays a set of 10 questions to the students one-by-one and collects user inputs.
- The questions may be chemical structure-based or chemical reaction-based. Either the student would pick an answer out of the 4 options available or is asked to draw a structure to answer the question respectively.
- After answering all the questions, the student is redirected to the results page.

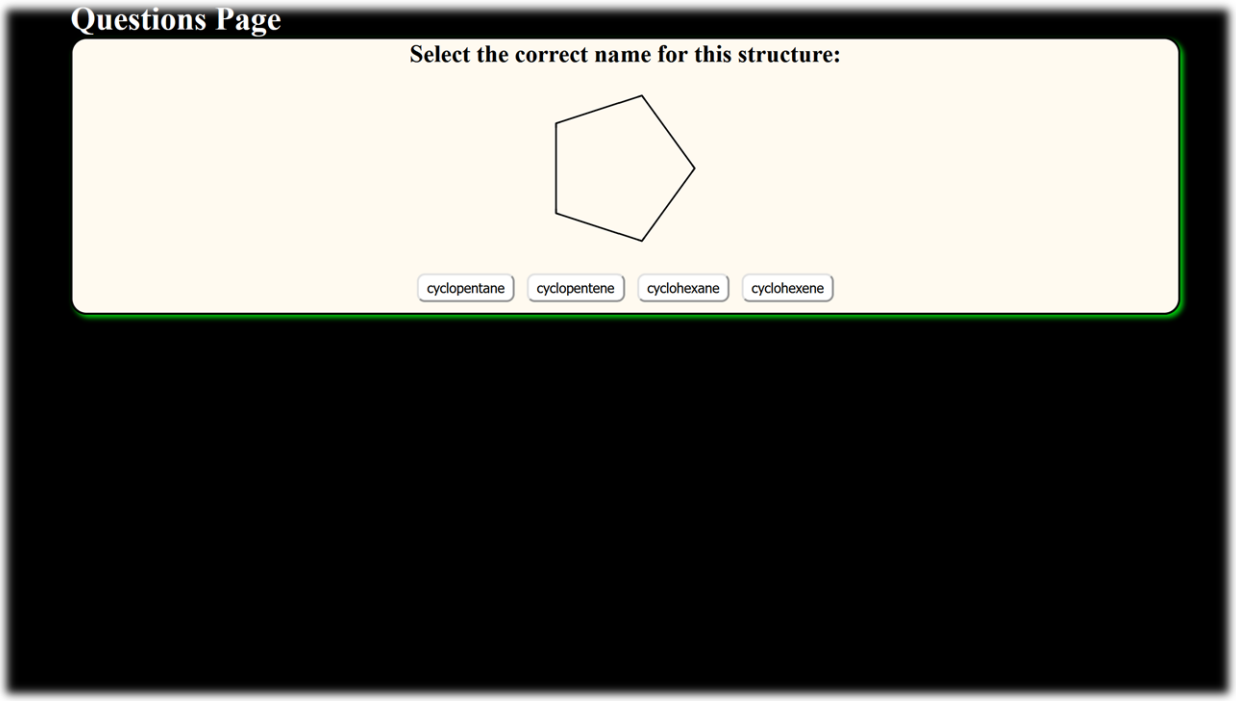


Fig. 3: User Interface of Questions page for a Structure Question

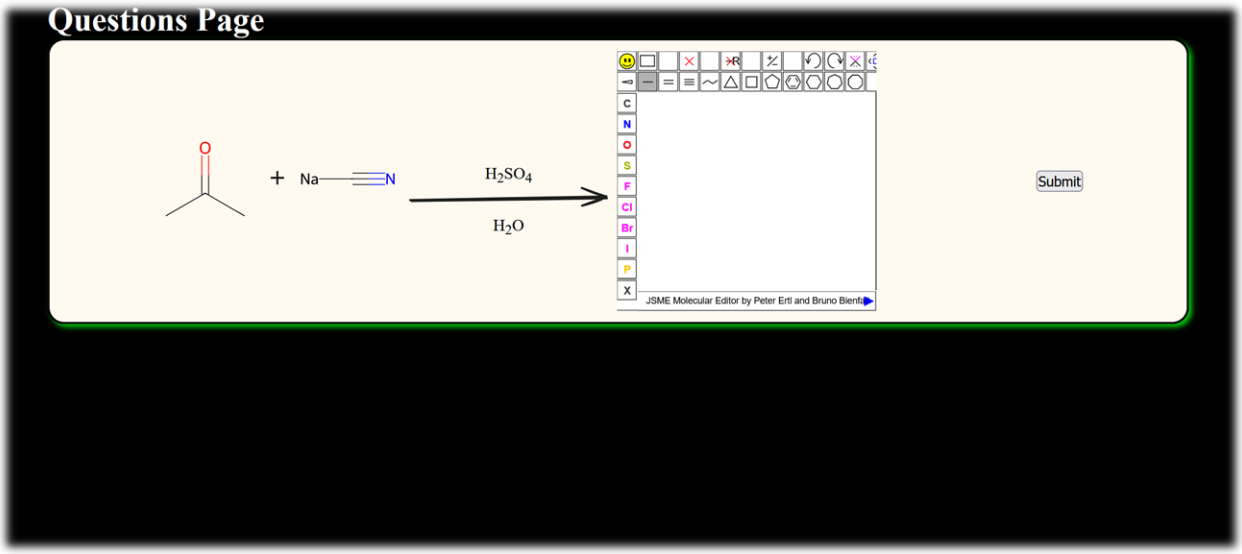


Fig. 4: User Interface of Questions page for a Reaction Question

4.4 Results Page:

- The page displays the questions, the student's answers, and the correct answers for the questions asked, as well as the points awarded for each one.
- The user is also shown a custom feedback message based on the score they have secured in the current attempt.
- The page also has a Return Button. When this button is clicked, the student is navigated back to the welcome page, from where they can either choose to play the quiz again or can logout of the application.

- Before displaying the data to the user, the score is stored and player statistics and scores are retrieved from the database in preparation for being displayed when the user returns back to welcome page

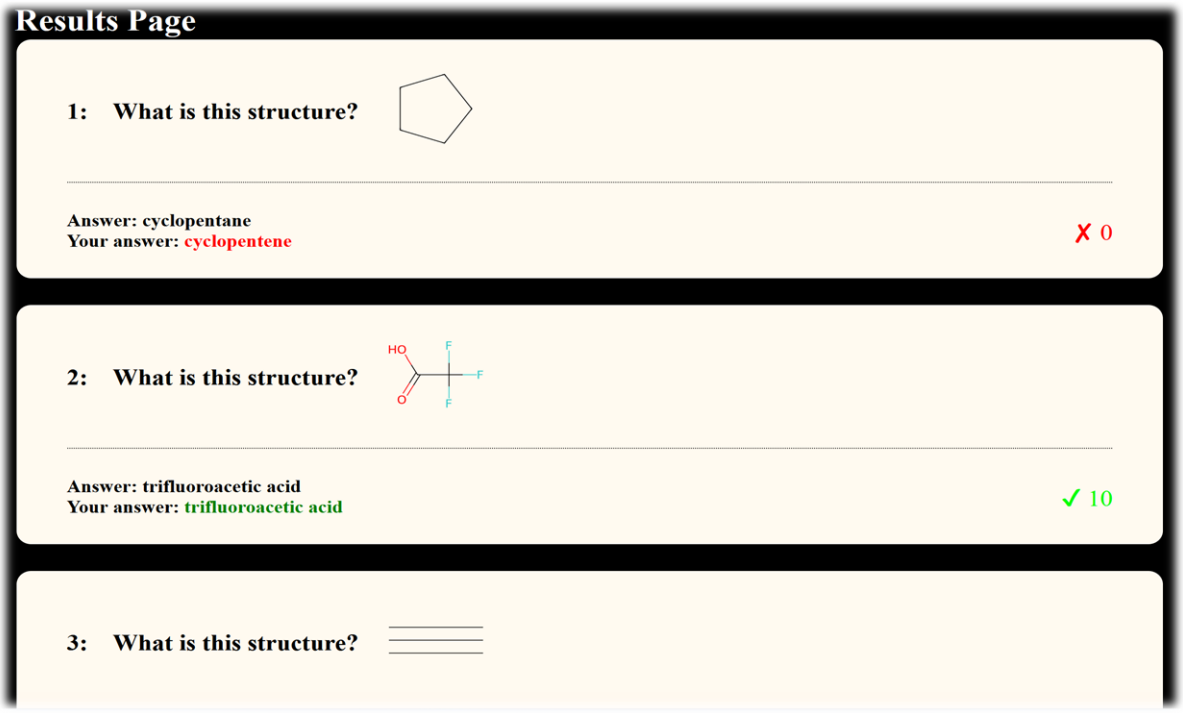


Fig. 5: User Interface of Results page

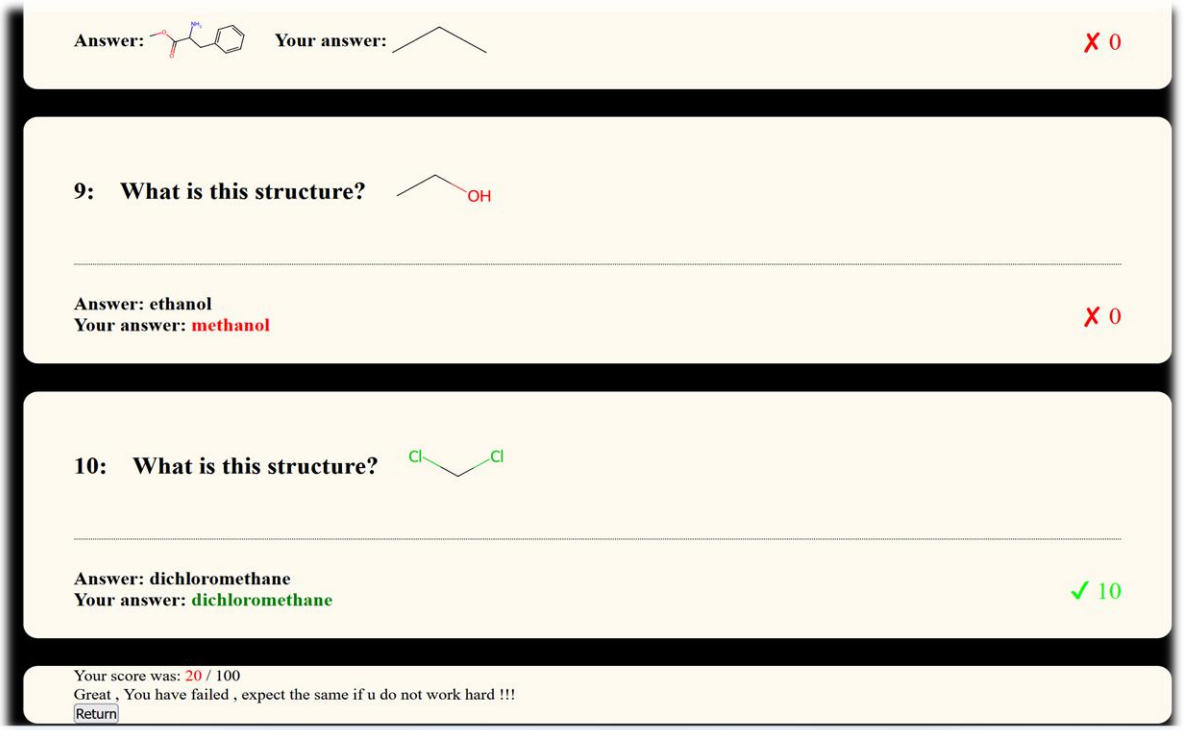


Fig. 6: User Interface of Results page with the scores and 'Return' button

4.5 Admin Page:

- This page enables an Admin to modify the data stored in the database in a user-friendly manner.
- The admin has access to the Users table, where they can add new users, modify the username or password of existing ones, and delete users.
- The admin has access to the Scores table, where they can modify the score of students' quiz attempts or delete the record entirely.
- The admin is provided the functionality for creating new questions of various types, as well as modifying or deleting existing questions.

Fig. 7: User Interface of Admin page with the scores and 'Return' button

5.Software Requirements and Design Specifications for Technology Usage

5.1 User-facing pages and their functionalities

5.1.1 Login Page

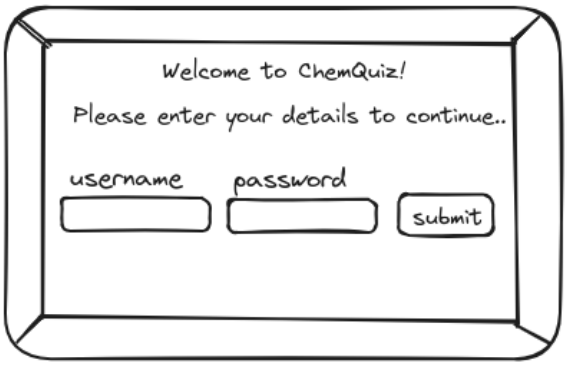


Fig. 8: Example user interface sketch for the login page

- Upon clicking the submit button, and after client-side validation, the entered username and password are sent to the server.
- The server sanitises the input, then queries the database using the username, and finally retrieves a hashed version of the user's password. The server hashes the entered password and compares it with the stored version.
- If the two don't match, or if the username was not located on the database, a failure response will be sent to the client, and the user will be prevented from proceeding. They will be prompted to retry. **(Prettyman, 2020).**
- If the credentials match, then the user has been validated. If the user is an admin, they will be directed to the administrative area. However, if the user is a student, the server will retrieve the score results of the student's 5 most recent quiz attempts, and a Leaderboard, via SQL queries. This data is sent to the client as JSON, stored in the session storage and then displayed to the user on the Welcome page.

5.1.2 Welcome Page

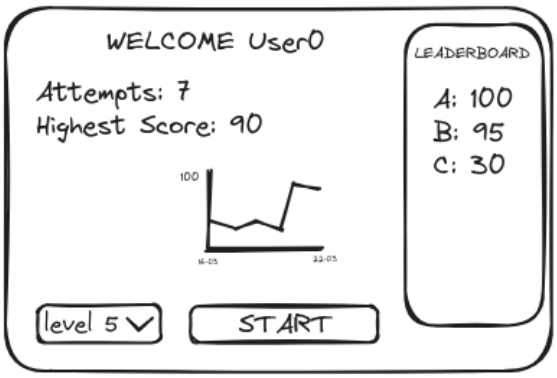


Fig. 9: Example user interface sketch for the welcome page

- After clicking the start button, the selected difficulty is sent to the server where it is incorporated in a SQL query to the database, retrieving 10 questions of appropriate difficulty.
- The questions are sent to the client in one batch as JSON for rendering the questions pages, preventing the need for network requests after each response. **(Smith, 2015).**

5.1.3 Questions Pages

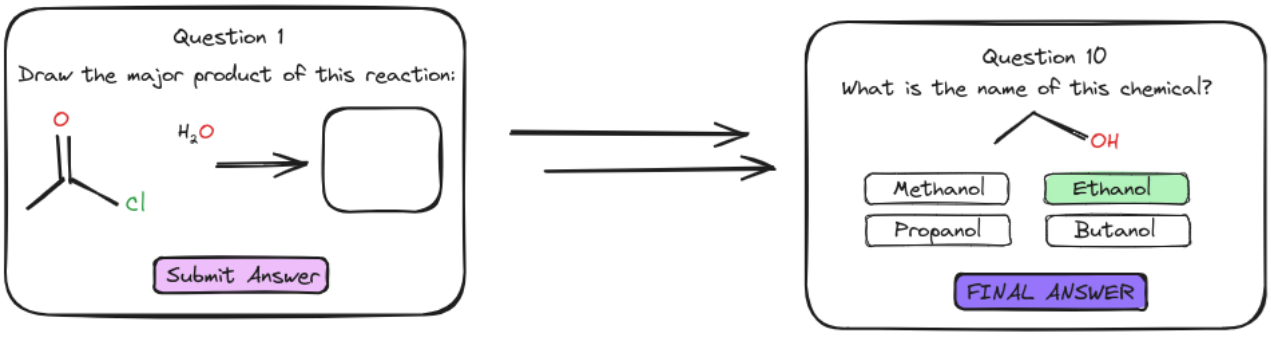


Fig. 10: Example user interface sketch for question pages 1 and 10

- Once a question has been answered, it gets stored as an array, and the next question is dynamically rendered to the screen using JavaScript. **(Simpson, 2023).**
- Once an answer to the final question is submitted, all responses in the array are sent to the server, alongside an identifier for the student.

- The server sends a SQL query to the database, retrieving the correct answers to all questions asked. It determines the student's score, and stores on the database using a SQL update query.
- The questions, answers and scores are sent to the client as JSON.

5.1.4 Results Page

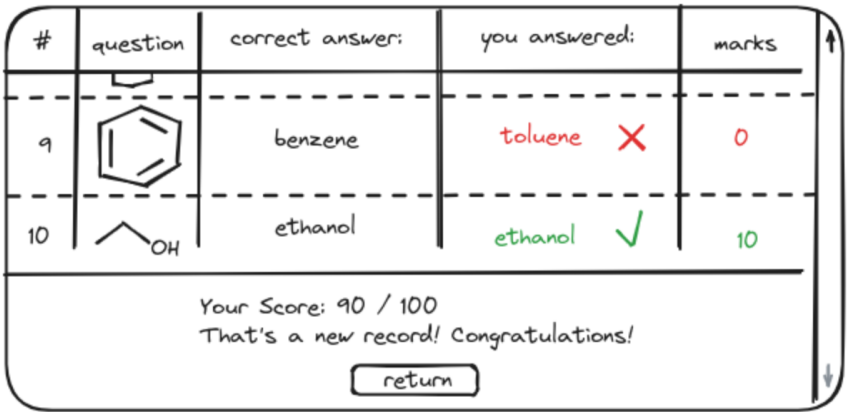


Fig. 11: Example user interface sketch for the results page

- Clicking the return button will cause a re-render of the screen to the welcome page, using the updated scores information that was retrieved earlier.

5.1.5 Admin Page

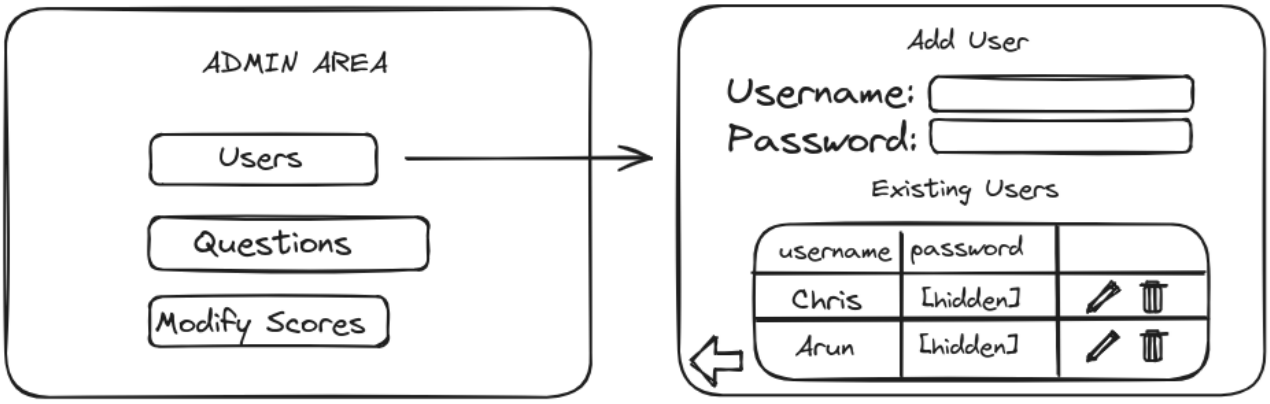


Fig. 12: User interface sketch of results page, with the outcome of clicking the "Users" button

- Clicking a button corresponding to a table in the database triggers the server to retrieve all records of that particular table. These are sent to the client as JSON, the user interface is rendered to the screen consisting of an "add item" section and "existing items" section.
- Filling in the input fields in the "add item" section and attempting to submit the data will cause client-side validation, and if successful, the data will be passed to the server which sanitises it. A SQL update query is performed, and a success (or failure) message in the form of JSON is sent to the client based on the result of the procedure.
- Clicking a button to modify the item will provide an interface similar to the add item section, however the fields will be rendered containing the existing information. The user may make changes and submit the data, in which case a procedure resembling "add item" is performed. **(Gehani, 2011).**
- Clicking the button to delete a record on the database will send the unique identifier for the record to the server, and a SQL delete query will remove it from the appropriate table.

5.2 Program Execution Flow Diagrams

5.2.1 Program execution flow diagram: Student

Fig. 13: Step-by-step description of the flow of execution of the program for students

5.2.2 Program execution flow diagram: Admin

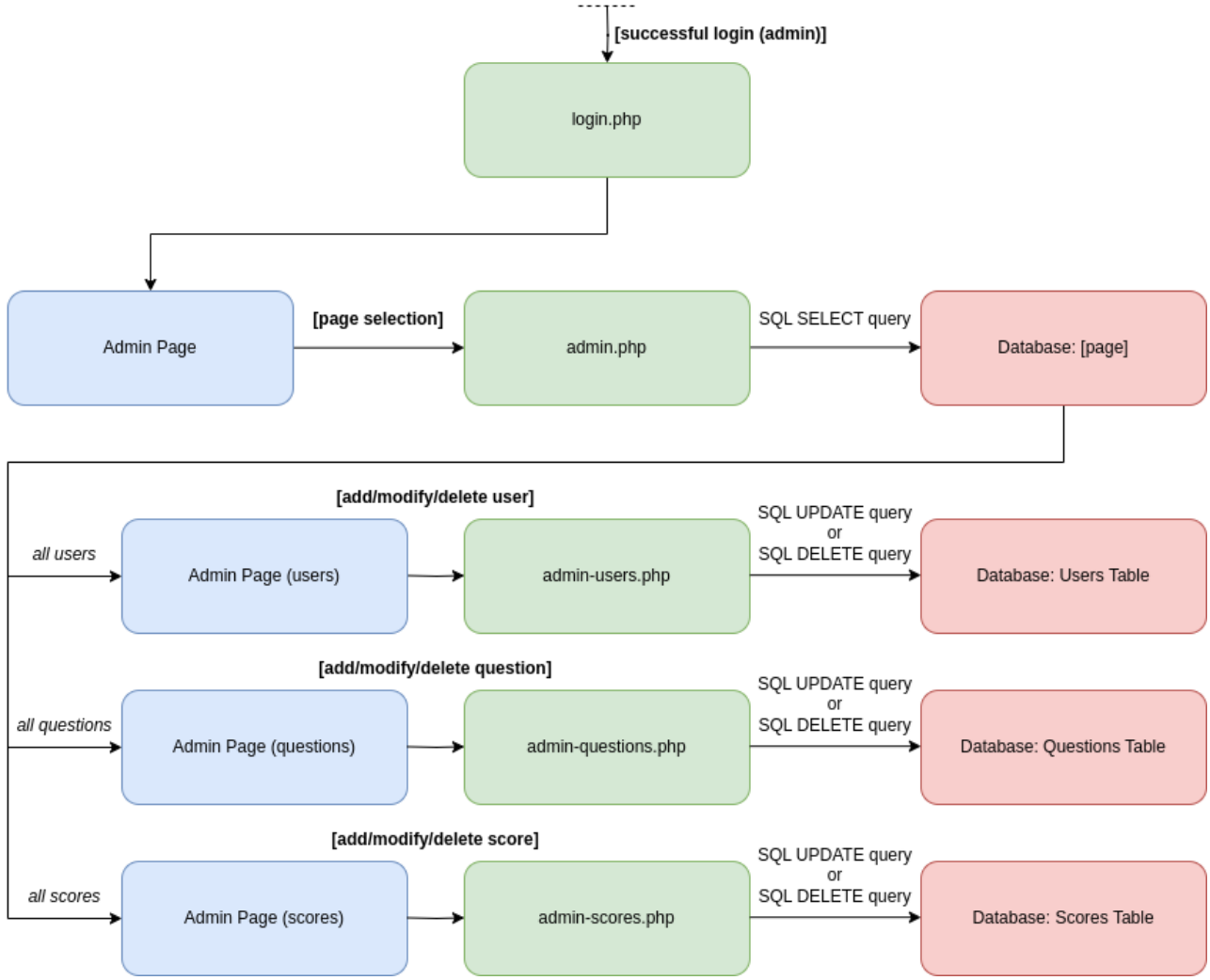


Fig. 14: Step-by-step description of the flow of execution of the program for administrators

5.3 Databases

A detailed database system with a list of associated tables is necessary to store, manage, and manipulate all essential data. To aid in this process, the following entity - relationship diagram was created. **(Delisle, 2006).**

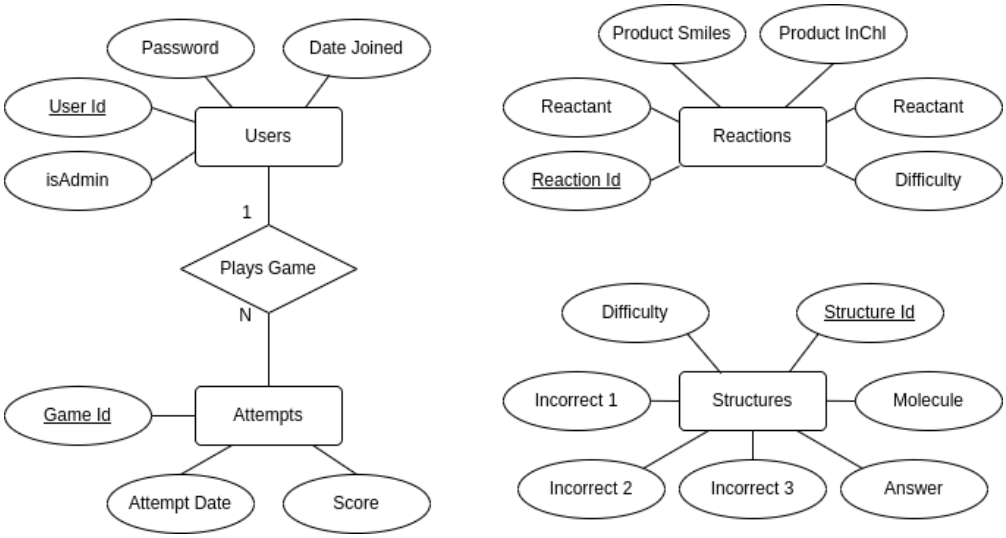


Fig. 15: Entity-relationship diagram describing the data involved in the proposed database

5.3.1 Users Table

- Stores the username, hashed password and creation date of each user.
- Each record has a user ID which uniquely identifies that user.

User Id	User Name	Password	Date-Joined	isAdmin
5	Arun	$2y$10$od4/W8	23.02.2024	false

Table 1: DB schema and an example record of USERS table. User Id is the primary key

5.3.2 Questions Tables

Each type of question will require its own table, containing information about the question to be asked, the correct answer, and a difficulty rating (see **Table 2** and **Table 3** below).

- Multiple-choice questions will include pre-chosen decoy answers.
- Each question will contain a unique ID to identify each question on each table.

Structural information about chemicals will be stored in two established formats:

- SMILES (simplified molecular-input line-entry system): a way to represent chemicals in string format.
    - For example, sodium chloride could be represented as "[Na+].[Cl-]".
    - SMILES can be transformed into SVGs (image format) on the client using a library called RDKit.js**. (Landrum, 2013).**
- InChI (International Chemical Identifier): a more recent method than SMILES to represent chemicals in string format. Unlike SMILES, chemicals in the InChI format are represented in a completely unambiguous manner.

- For example, sodium chloride is "InChI=1S/ClH.Na/h1H;/q;+1/p-1" in the InChI format, whereas both "[Na+].[Cl-]" and "[Cl-].[Na+]" are valid SMILES.
- To compare with the stored InChI string answer to a quiz question, a user's inputted chemical structure input needs to be converted the InChI format using RDKit.js.

Reaction Id	Reactant	Reagent	Product Smiles	Product Inchi	Difficulty
5	CC(=O)Cl	CCCCN	CCCCNC(C)=O	InChI=1S/C6H13NO/c1-3-4-5-7-6(2)8/h3-5H2,1-2H3,(H,7,8)	2

Table 2: DB schema and an example record of REACTIONS table. Reaction Id is the primary key

Structure Id	Molecule	Answer	Incorrect 1	Incorrect 2	Incorrect 3	Difficulty
3	Cn1cnc2c1c(=O)n(C)c(=O)n2C	caffeine	ethanol	morphine	aspirin	4

Table 3: DB schema and an example record of STRUCTURES table. Structure Id is the primary key

5.3.3 Scores Table

- Contains details about the scores a specific user makes for each of their quiz attempts – the user Id corresponding to an entry in the Users table, the total score and the attempt date.
- Every record has a Game Id to uniquely identify each quiz attempt. **(Delisle, 2006).**

Game Id	User Id	Score	Attempt Date
1	5	8	02.04.2024
2	5	9	04.04.2024

Table 4: DB schema and an example record of SCORES table. Game Id is a primary key, User Id is a foreign key referencing User Id in the USERS table

6. Appendix

6.1 Installation Guide

6.1.1: On the web

The application has been deployed on the web, and can be accessed though the following url on any web browser: https://cropie.online

6.1.2: Local Installation

3 steps are required for running the program locally. They are:

1) Decompress/unzip the source code to an appropriate folder.
2) Installation of XAMPP, which includes MariaDB and PHP.
3) Populating the database.

6.1.2.1: Decompress/unzip the source code

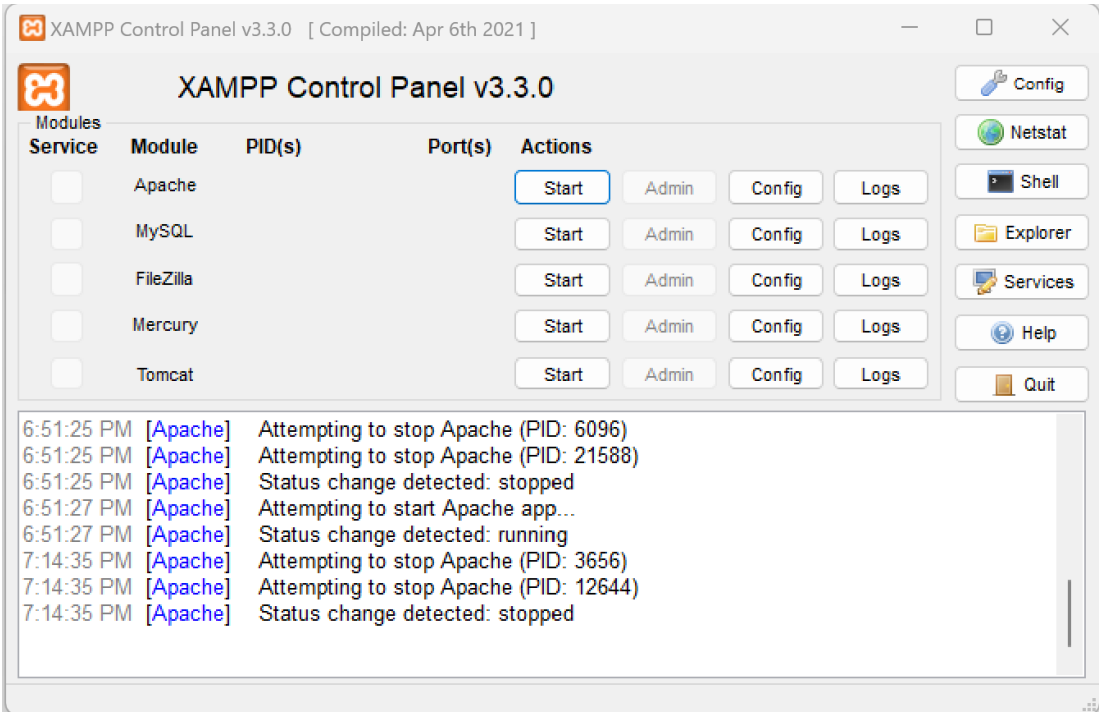
Please unzip the source code in an appropriate folder, for example /chemquiz. The files are provided as-is – there is no compilation required.

6.1.2.2: Setting up the Web Server

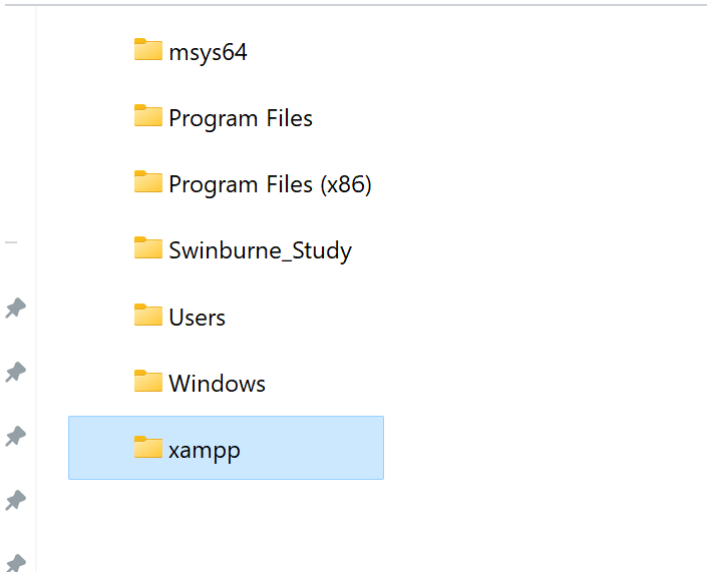
Chemquiz uses a web server to serve the application files to web browsers, and to provide a connection between the browser (client) and database. XAMPP is an open-source web server package developed by Apache, which can be used as a PHP server on a local computer.

1. Download the latest version of the XAMPP server pack from https://www.apachefriends.org/.

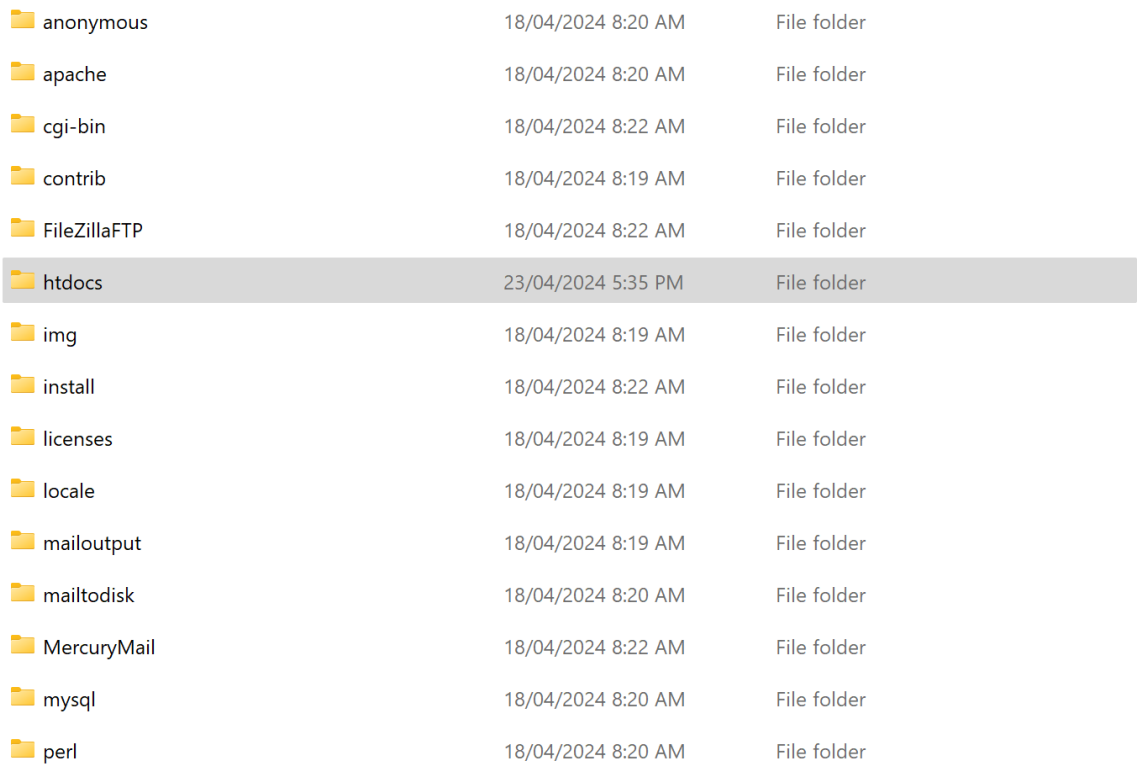
2. Run the executable installation wizard and set up the XAMPP server along with the XAMPP control panel being selected.



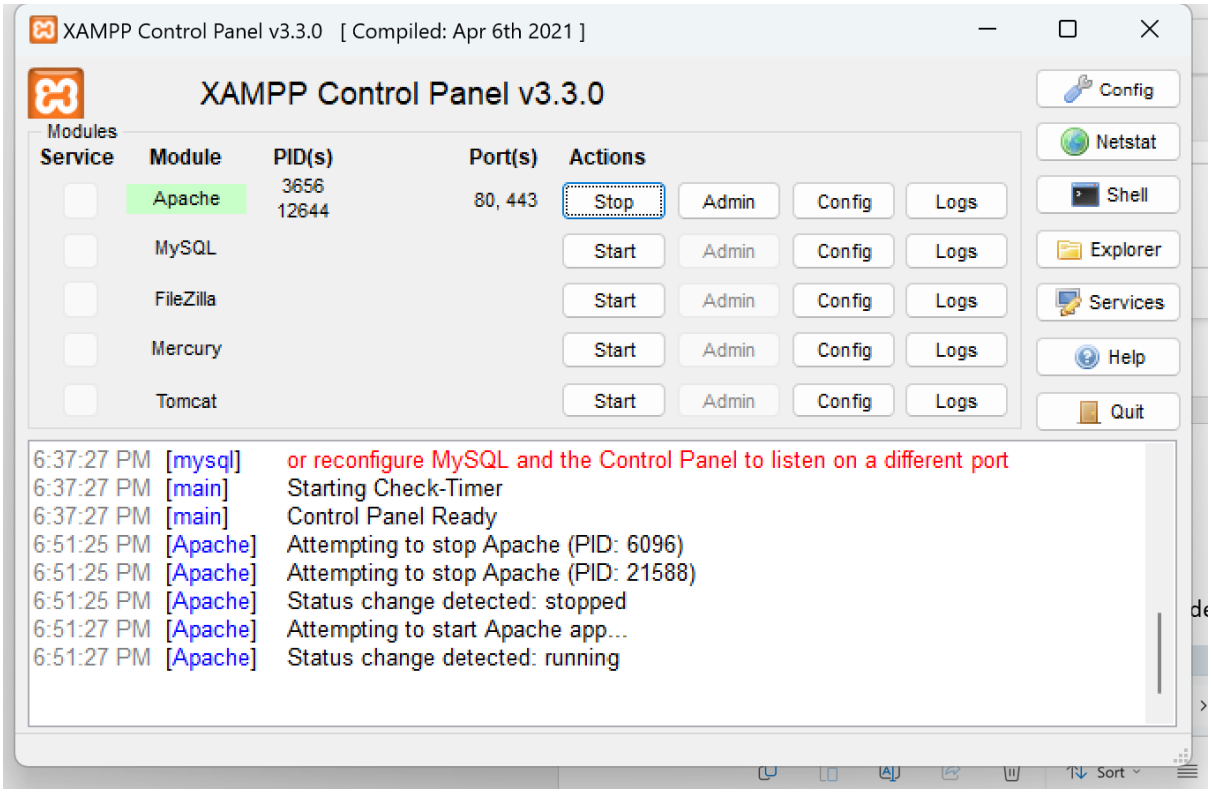
3. Please go to the place where the XAMPP root folder is placed. It is likely inside the C drive where the program files are generally kept.



4. Click and go inside the XAMPP folder now and locate 'htdocs'. The chemquiz directory, containing the unzipped program files, must be transferred to htdocs folder.



5. With the files in place, it is time to start the APACHE web server and the MySQL server. Open the XAMPP control panel and start the servers using the start button



6. Please note that if a SQL server is already installed and running in your localhost (PORT :3306), then starting the MySQL Server of XAMPP would lead to an error. In this situation there is no need to start the XAMPP MySQL server.

6.1.2.3: Setting up the Database

If not already installed through XAMPP, please follow the installation procedures for either MariaDB (https://mariadb.org/) or MySQL (https://www.mysql.com/) for your particular operating system. These open-source databases conveniently integrate with PHP, and in this program will be used to store the user, scores, and questions information.

Open mariadb or mysql, and enter the following command:

```
CREATE DATABASE Chemquiz;
```

Populate the database by either manually entering the SQL queries in populatedatabase.sql, or by running the following command in the root directory of the chemquiz program:

```
mariadb Chemquiz < populatedatabase.sql
```

or

```
mysql Chemquiz < populatedatabase.sql
```

After the database has been populated with the default information, check to see if the data matches these pictures, by running the following commands in the database:

```
USE DATABASE Chemquiz;
```

SHOW TABLES;

```
MariaDB [Chemquiz]> SHOW TABLES;
```

Tables_in_Chemquiz
ReactionQ
Scores
StructureQ
Users

4 rows in set (0.001 sec)

SELECT * FROM Users;

```
MariaDB [Chemquiz]> SELECT * FROM Users;
```

userId	username	password	dateJoined	isAdmin
1	student	$2y$10$0DTKrN1.96bSniLN85bQoeNjF8bVEUsMAKlhyxg9P/GWxn5gDQGr.	2024-05-17	0
2	admin	$2y$10$0DTKrN1.96bSniLN85bQoeNjF8bVEUsMAKlhyxg9P/GWxn5gDQGr.	2024-05-17	1

2 rows in set (0.001 sec)

SELECT * FROM Scores;

```
MariaDB [Chemquiz]> SELECT * FROM Scores;
Empty set (0.001 sec)
```

SELECT * FROM StructureQ;

```
MariaDB [Chemquiz]> SELECT * FROM StructureQ;
```

structureId	molecule	answer	incorrect1	incorrect2	incorrect3	difficulty
1	c1ccccc1	benzene	toluene	methanol	ethanol	0
2	CCO	ethanol	methanol	propanol	tert-butanol	0
3	CN	methylamine	ethylamine	propylamine	butylamine	0
4	C1CCCC1	cyclopentane	cyclopentene	cyclohexane	cyclohexene	0
5	CN4CC[C@]23c1c5ccc(O)c1O[C@H]2[C@@H](O)C=C[C@H]3[C@H]4C5	morphine	aspirin	codeine	heroin	1
6	CC(=O)Oc1ccc2C[C@@H]5[C@@H]3C=C[C@H](OC(C)=O)[C@@H]4Oc1c2[C@]34CCN5C	heroin	morphine	oxycodone	cocaine	1
7	COC(=O)[C@H]2[C@@H](OC(=O)c1ccccc1)C[C@@H]3CC[C@H]2N3C	cocaine	dopamine	caffeine	ethanol	1
8	Cn1cnc2c1c(=O)n(C)c(=O)n2C	caffeine	ethanol	cocaine	heroin	1
9	ClCCl	dichloromethane	chloromethane	chloroform	carbon tetrachloride	0
10	C#C	ethyne	alkyne	alkene	alkane	0
11	C[C@@H](N)C(=O)O	alanine	glycine	phenylalanine	proline	0
12	c1ccncc1	pyridine	pyrimidine	pyrine	2,6-lutidine	0
13	O=C(O)C(F)(F)F	trifluoroacetic acid	acetic acid	ethanoic acid	triflic acid	0
14	CC(C)(C)S	tert-butylthiol	tert-butanol	n-butanol	n-butylthiol	0
15	O=Cc1ccccc1	benzaldehyde	benzyl alcohol	benzoic acid	benzoin	0
16	CC(=O)Oc1ccccc1C(=O)O	aspirin	paracetamol	ibuprofen	methyl salicylate	1
17	Cc1c(N(=O)=O)cc(N(=O)=O)cc1N(=O)=O	trinitrotoluene	dinitrotoluene	nitroglycerine	acetone peroxide	1
18	COP(=O)(OC)C(=N#N)C(C)=O	Ohira-Bestmann reagent	Corey-Fuchs reagent	Shibasaki-Kumagai reagent	HWE reagent	1
19	COc1cc(CNC(=O)CCCC/C=C/C(C)C)ccc1O	capsaicin	mustard gas	nonivamide	allicin	1
20	NCCc1c[nH]c2ccc(O)cc12	serotonin	methamphetamine	tryptophan	safrole	1
21	CN1CCC[C@H]1c2cccnc2	nicotine	nylon	napthylene	normorphine	1

21 rows in set (0.001 sec)

SELECT * FROM ReactionQ;

```
MariaDB [Chemquiz]> SELECT * FROM ReactionQ;
```

reactionId	reactant	reagent	productSmile
1	CC(=O)O	NCc1ccccc1	CC(=O)NCc1ccccc1
2	CC(C)=O	N#C[Na]	CC(C)(O)C#N
3	O=C1CCCCC1		OC1CCCCC1
4	C/C(C)=C(C)C	BrBr	CC(C)(Br)C(C)(C)Br
5	c1ccccc1	BrBr	Brc1ccccc1
6	CCC(C)=O	Br[Mg]c1ccccc1	CCC(C)(O)c1ccccc1
7	COCOc1cccc(C)c1	O=C=O	COCOc1cc(C)ccc1C(=O)O
8	CC(=O)Cc1ccccc1	CN	CNC(C)Cc1ccccc1
9	C=CC=C	COc1cc(=O)c(C)cc1=O	COC2=CC(=O)C1(C)CC=CCC1(C)C2=O
10	CCCC#C[Si](C)(C)C(C)(C)C		C#CCCC
11	COC(=O)C(Cc1ccccc1)NC(=O)OC(C)(C)C		COC(=O)C(N)Cc1ccccc1
12	CC1(C)CC(=O)CC(=O)C1	O=S(=O)(OS(=O)(=O)C(F)(F)F)C(F)(F)F	CC1(C)CC(=O)C=C(OS(=O)(=O)C(F)(F)F)C1
13	O=c2c1ccccc1c3ccccc23	NN	NN=c2c1ccccc1c3ccccc23
14	Nc1ccccc1Br	OB(O)c1ccccc1	Nc1ccccc1c2ccccc2
15	N#CCc1ccccc1		CCOC(=O)Cc1ccccc1
16	Oc2ccc1ccccc1c2	O=NO[Na]	O=Nc1c(O)ccc2ccccc12
17	Br/C=C/c1ccccc1		C#Cc1ccccc1
18	O=c1oc(=O)c2ccccc12		O=c1[nH]c(=O)c2ccccc12
19	Cc1cc(C)c(N)c(C)c1	O=CC=O	Cc2cc(C)c(/N=C/C=N/c1c(C)cc(C)cc1C)c(C)c2
20	C[Si](C)(C)c1ccccc1OS(=O)(=O)C(F)(F)F	CCOC(=O)C=N#N	CCOC(=O)c1n[nH]c2ccccc12

reactionId	productInchi
1	InChI=1S/C9H11NO/c1-8(11)10-7-9-5-3-2-4-6-9/h2-6H,7H2,1H3,(H,10,11)
2	InChI=1S/C4H7NO/c1-4(2,6)3-5/h6H,1-2H3
3	InChI=1S/C6H12O/c7-6-4-2-1-3-5-6/h6-7H,1-5H2
4	InChI=1S/C6H12Br2/c1-5(2,7)6(3,4)8/h1-4H3
5	InChI=1S/C6H5Br/c7-6-4-2-1-3-5-6/h1-5H
6	InChI=1S/C10H14O/c1-3-10(2,11)9-7-5-4-6-8-9/h4-8,11H,3H2,1-2H3
7	InChI=1S/C10H12O4/c1-7-3-4-8(10(11)12)9(5-7)14-6-13-2/h3-5H,6H2,1-2H3,(H,11,12)
8	InChI=1S/C10H15N/c1-9(11-2)8-10-6-4-3-5-7-10/h3-7,9,11H,8H2,1-2H3
9	InChI=1S/C13H16O3/c1-12-6-4-5-7-13(12,2)11(15)9(16-3)8-10(12)14/h4-5,8H,6-7H2,1-3H3
10	InChI=1S/C5H8/c1-3-5-4-2/h1H,4-5H2,2H3
11	InChI=1S/C10H13NO2/c1-13-10(12)9(11)7-8-5-3-2-4-6-8/h2-6,9H,7,11H2,1H3
12	InChI=1S/C9H11F3O4S/c1-8(2)4-6(13)3-7(5-8)16-17(14,15)9(10,11,12)/h3H,4-5H2,1-2H3
13	InChI=1S/C13H10N2/c14-15-13-11-7-3-1-5-9(11)10-6-2-4-8-12(10)13/h1-8H,14H2
14	InChI=1S/C12H11N/c13-12-9-5-4-8-11(12)10-6-2-1-3-7-10/h1-9H,13H2
15	InChI=1S/C10H12O2/c1-2-12-10(11)8-9-6-4-3-5-7-9/h3-7H,2,8H2,1H3
16	InChI=1S/C10H7NO2/c12-9-6-5-7-3-1-2-4-8(7)10(9)11-13/h1-6,12H
17	InChI=1S/C8H6/c1-2-8-6-4-3-5-7-8/h1,3-7H
18	InChI=1S/C8H5NO2/c10-7-5-3-1-2-4-6(5)8(11)9-7/h1-4H,(H,9,10,11)
19	InChI=1S/C20H24N2/c1-13-9-15(3)19(16(4)10-13)21-7-8-22-20-17(5)11-14(2)12-18(20)6/h7-12H,1-6H3/b21-7+,22-8+
20	InChI=1S/C10H10N2O2/c1-2-14-10(13)9-7-5-3-4-6-8(7)11-12-9/h3-6H,2H2,1H3,(H,11,12)

reactionId	catalyst	solvent	temperature	time	difficulty
1	PyBOP	THF	25	1	1
2	H2SO4	H2O	NULL	NULL	0
3	NaBH4	iPrOH	NULL	NULL	0
4			NULL	NULL	0
5	FeBr3		NULL	NULL	0
6		THF	0	1	0
7			NULL	NULL	1
8	Al/Hg		NULL	NULL	1
9		toluene	100	96	1
10	TBAF	THF	25	1	0
11	CF3COOH	DCM	25	1	0
12	C5H5N	DCM	-78	NULL	1
13		EtOH	-78	NULL	0
14	Pd(PPh3)4	toluene/EtOH/H2O	NULL	16	1
15	H2SO4	EtOH	78	6	0
16	H2SO4	H2O/NaOH	0	2	1
17	KOH		200	NULL	1
18	NH4OH	H2O	300	NULL	0
19		iPrOH, H2O	NULL	24	1
20	TBAF	THF	25	1	1

20 rows in set (0.001 sec)

The next step is to ensure that the database connection settings for the program are correct. Open the settings.php file in the root directory of the program with a text editor, and change the settings based on your personal mariadb / mysql settings:

Settings.php

```
<?php
        $host = 'localhost';
        $user = "<your username>";
        $pwd = "<your password>";
        $sql_db = "Chemquiz";
?>
```

The database has now been set up.

6.1.2.4: Running the program

With XAMPP installed, the source code present in the XAMPP/htdocs folder, and the database populated with data, the program should be ready for use. Each time the program is to be run, it is essential to ensure that XAMPP is open, and the Apache web server (and SQL server if required) is running.

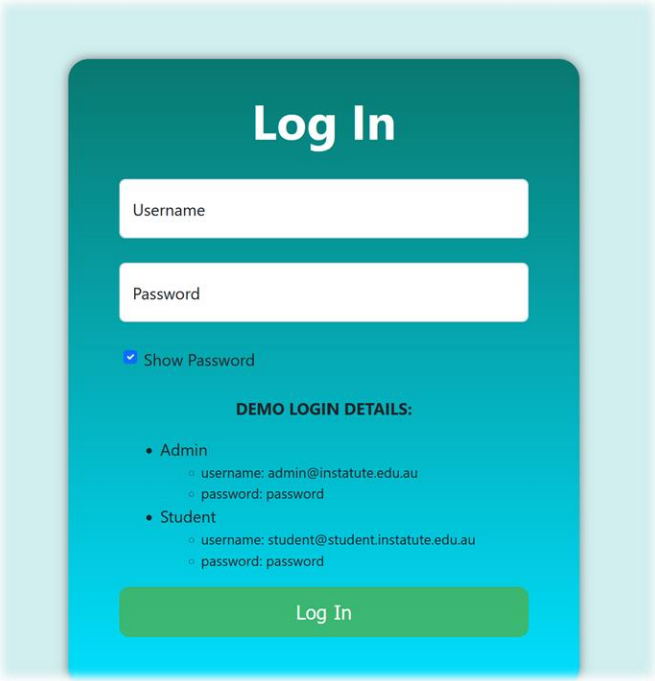
1. Please type ‘localhost’ in the URL of any browser of your choice. This opens the folders listed and placed under the ‘htdocs’ folder, which is displayed here through the XAMPP APACHE web server.

2. Click the main folder ‘chemquiz’. This starts the program execution on the server.

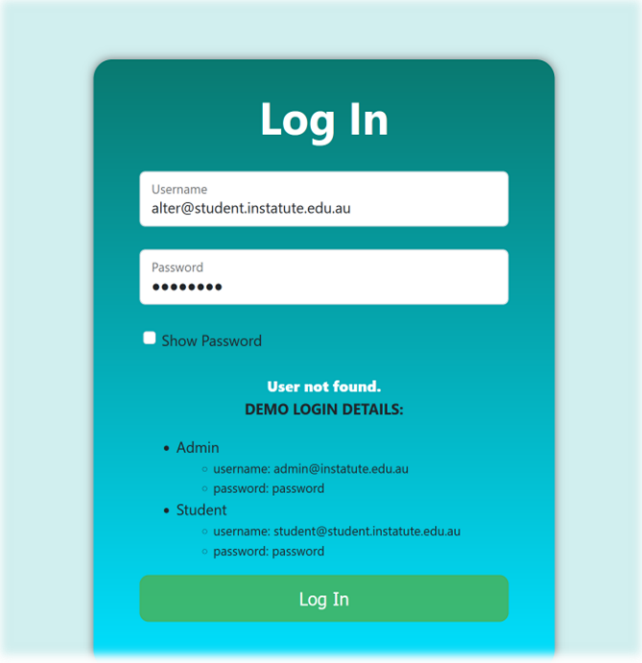
6.2 User Manual: Step by step guide to use the application:

6.2.1 Student User:

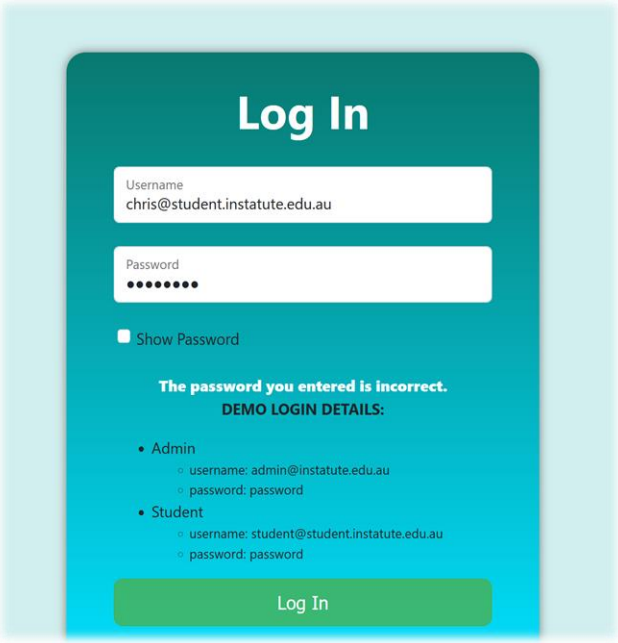
1. Please enter the User Name and password.



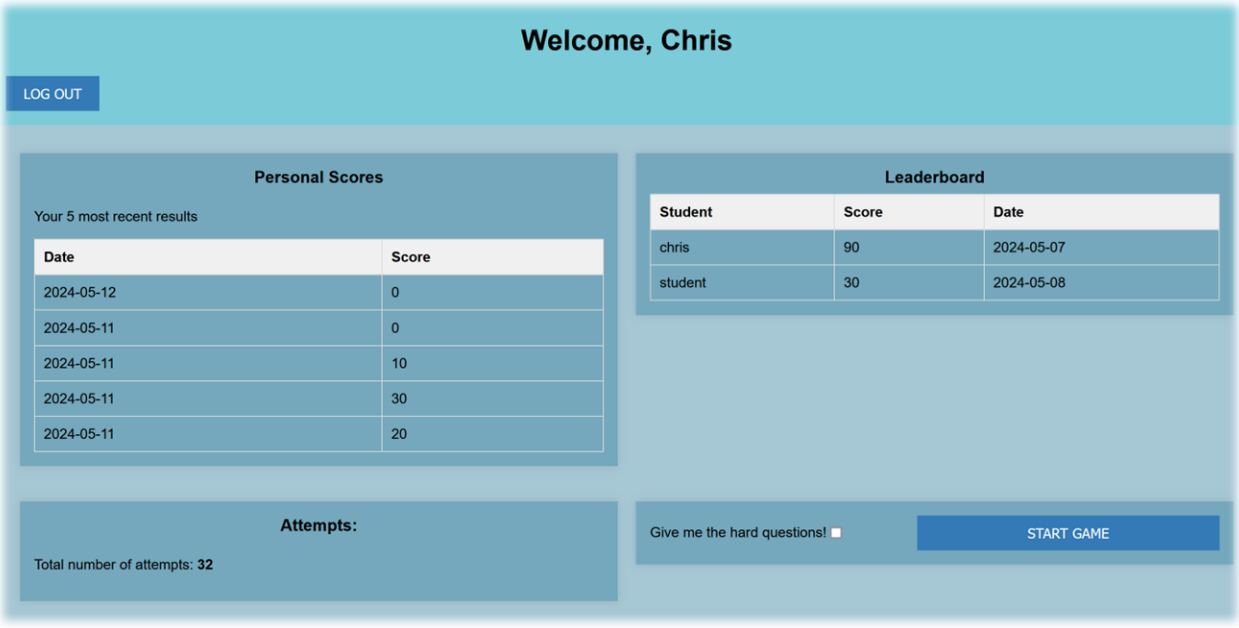
2. If user name is ‘incorrect’, user not found message will be displayed. Please retry.



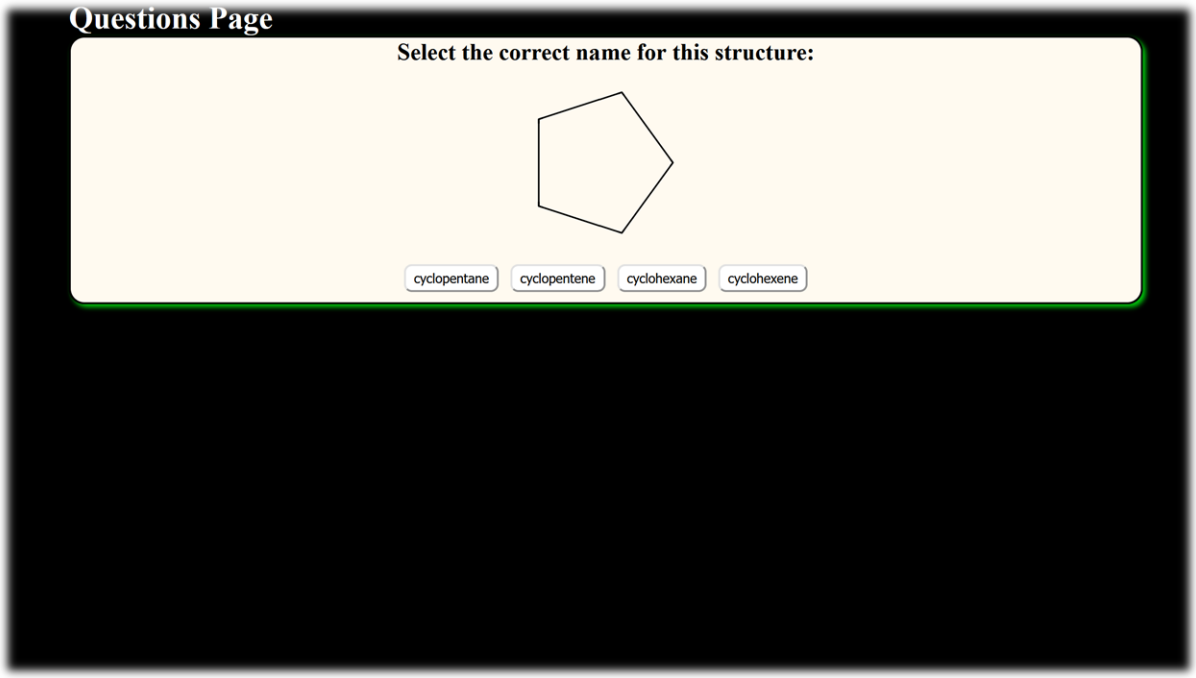
3. If the username is ‘correct’, but password is ‘incorrect’, password incorrect message is displayed.
Please retry.



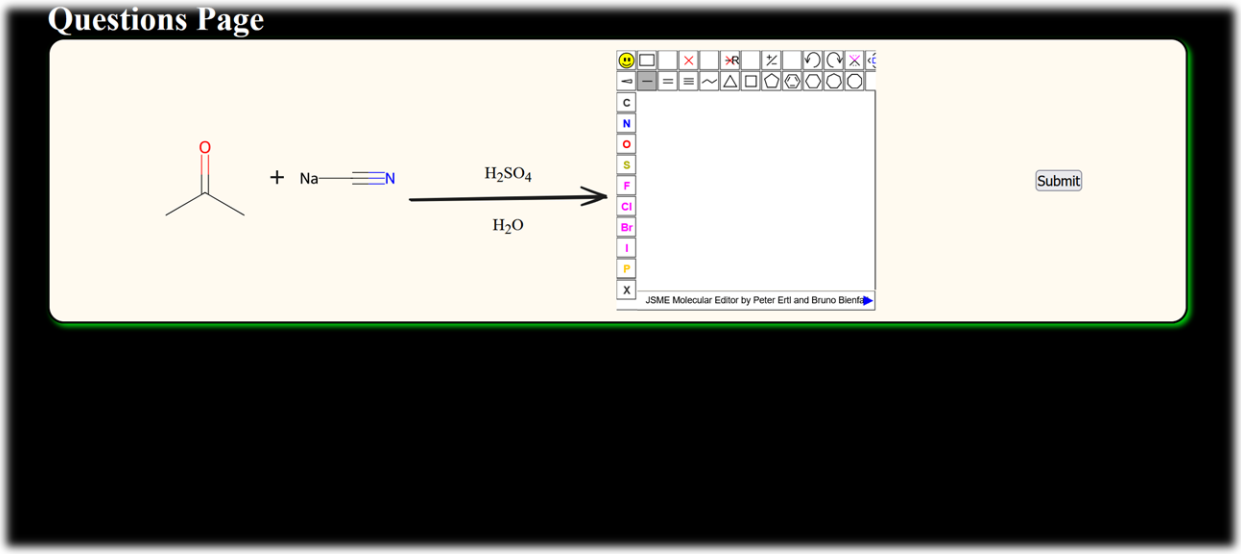
4. If Correct Username and password is entered, user is sent to the login page. Please select the check box for difficult questions on welcome page. Leave it unselected for questions of normal difficulty. Users can see their 5 most recent results as well as the student leaderboard. Press START GAME.



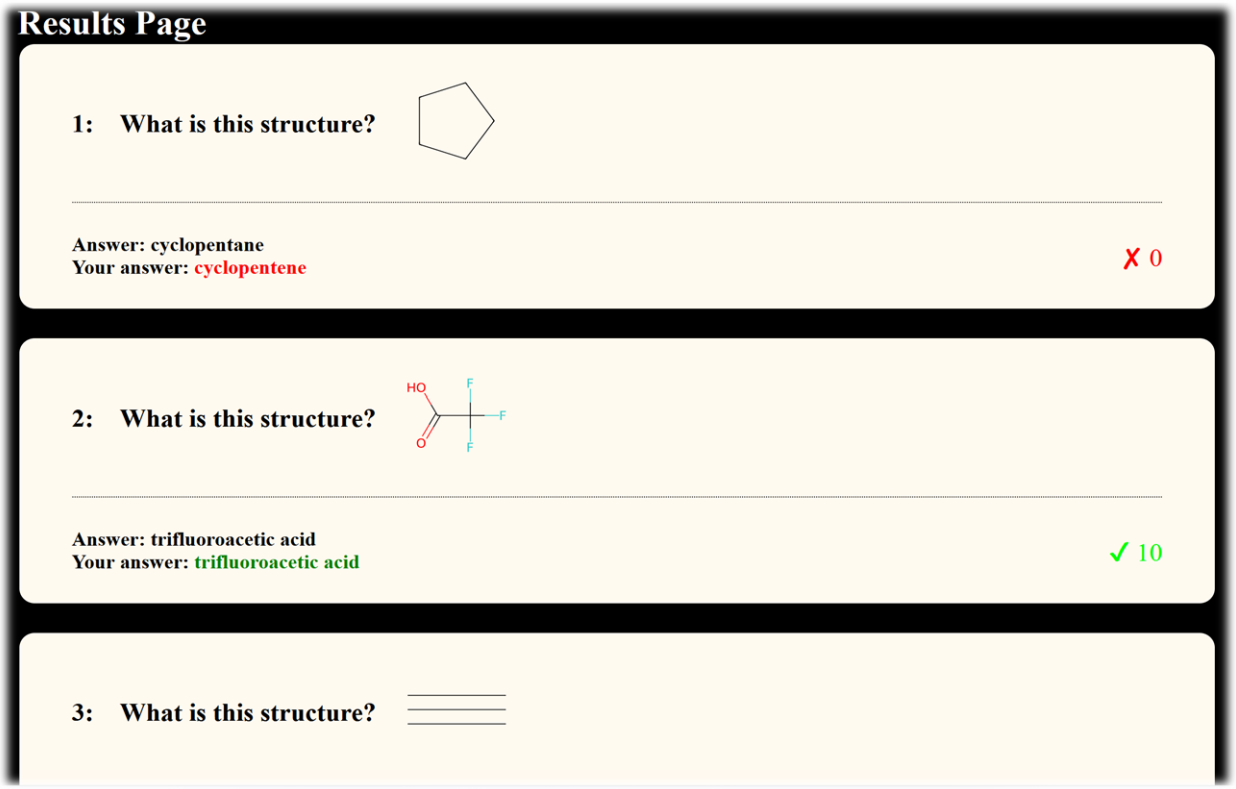
5.User is sent to Questions page. Select an answer if it’s a multi choice question, which will move the user to the next question.

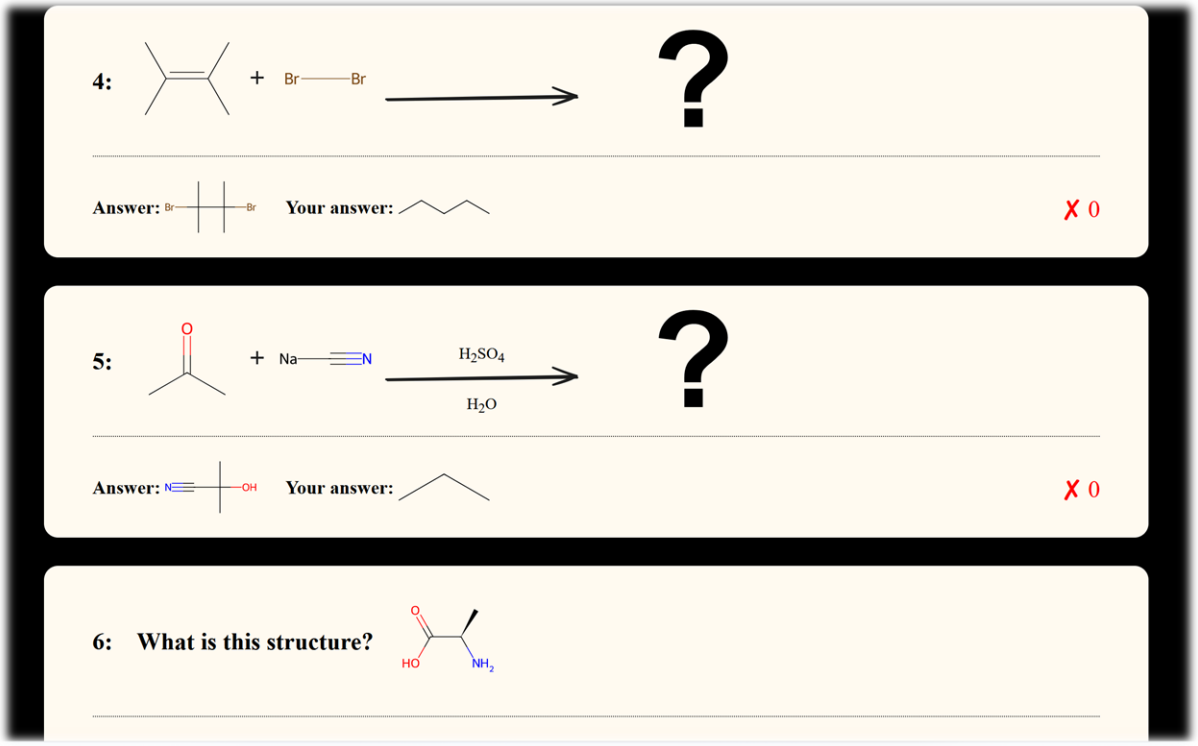


6. Draw a structure to answer the question if it is a reaction question. Click Submit when finished to move to the next question.

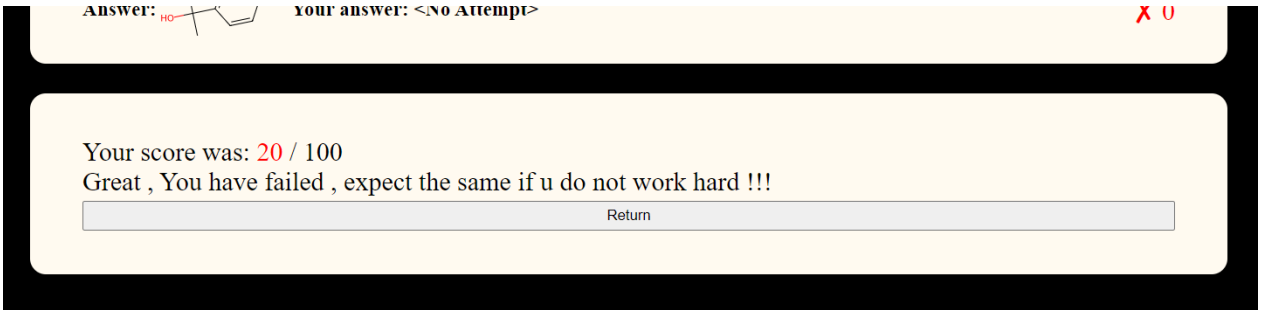


7. The user is routed to Results page automatically after answering 10 questions. They are shown the set of questions they were given, their answers and the correct answers, and the scores awarded.

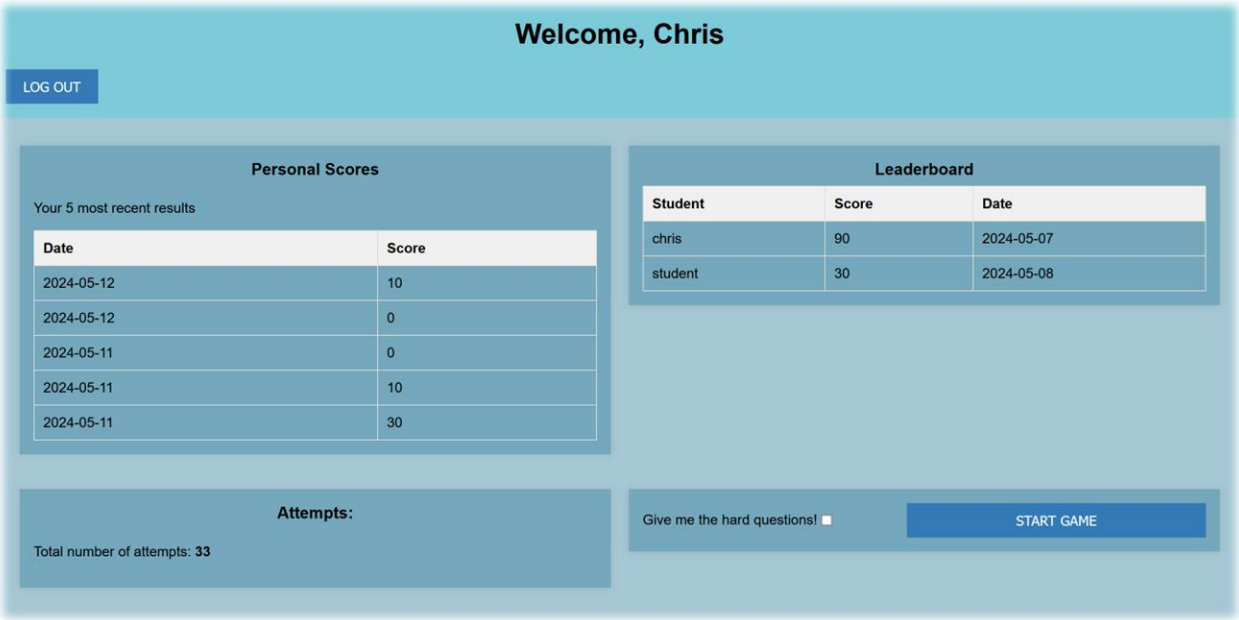




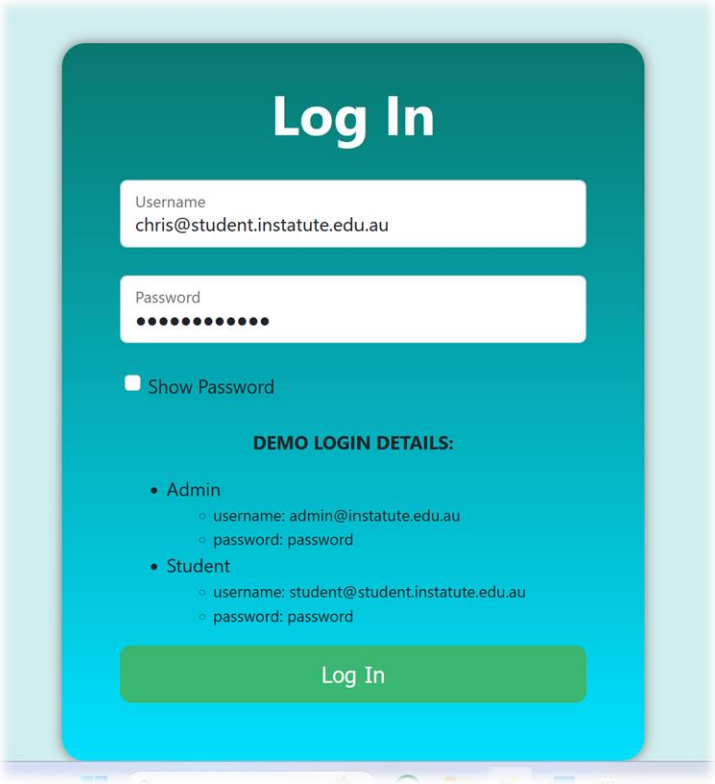
8. User is also shown their overall score and a custom message based on their performance.



9. On clicking RETURN button, user is navigated back to the welcome page. User can see their updated score and potentially an updated leaderboard. User can press START GAME to either start playing the quiz again (or) can press LOGOUT to exit the session.

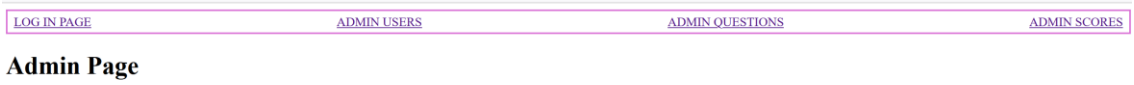


10. On pressing LOG OUT, the user is logged out of his/her session and end up on the Login page.



6.2.2 Admin User:

1. Upon entering an Admin username and password, an admin user is navigated to Admin page.



2. Admin user can choose ‘ADMIN USERS’ option to
   - ADD a new admin user
   - DELETE or EDIT details of an existing admin user

BACK ADMIN USERS

Admin Users Page

Add New User

Username Password Admin? Add User

Table of Existing Users

id	username	date joined	admin?	EDIT	PASS	DEL
30	chris	2024-05-07	no	E	P	X
34	admin	2024-05-08	yes	E	P	X
36	student	2024-05-08	no	E	P	X
39	Arun	2024-05-11	yes	E	P	X

3. Admin chooses to add a new user.

BACK ADMIN USERS

Admin Users Page

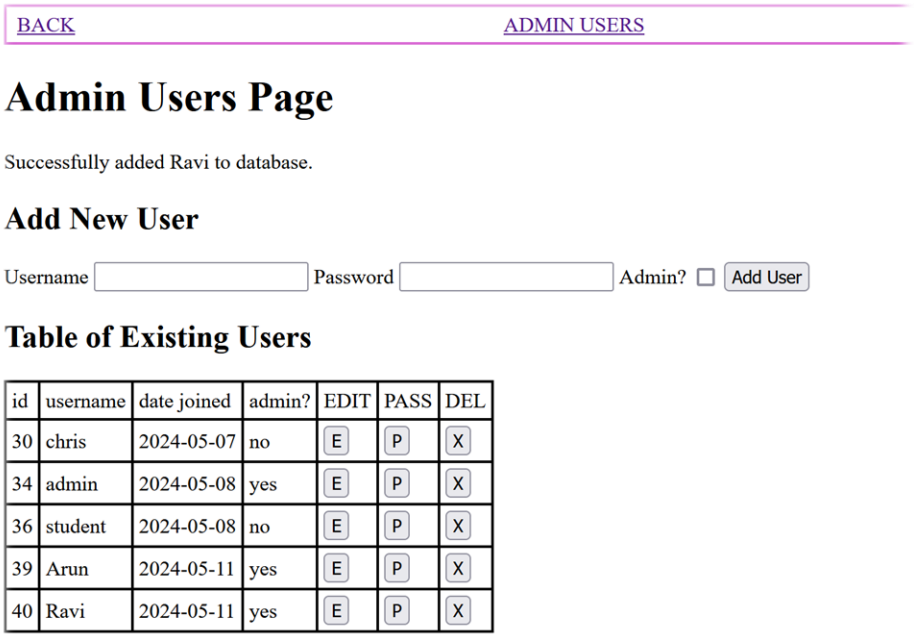
Add New User

Username Ravi Password Ravi@1457 Admin? Add User

Table of Existing Users

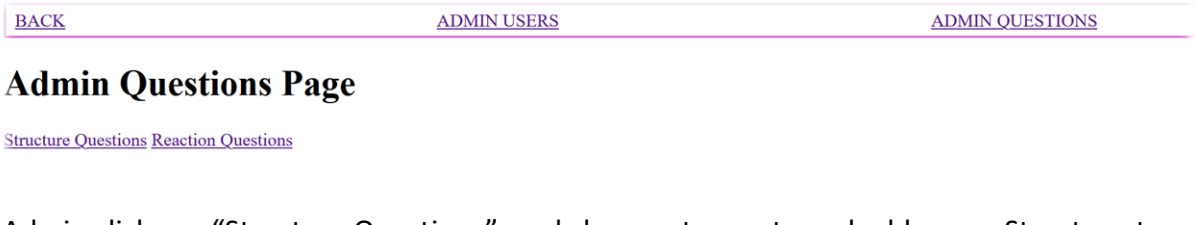
id	username	date joined	admin?	EDIT	PASS	DEL
30	chris	2024-05-07	no	E	P	X
34	admin	2024-05-08	yes	E	P	X
36	student	2024-05-08	no	E	P	X
39	Arun	2024-05-11	yes	E	P	X

4.New user added and updated in the Users Database.

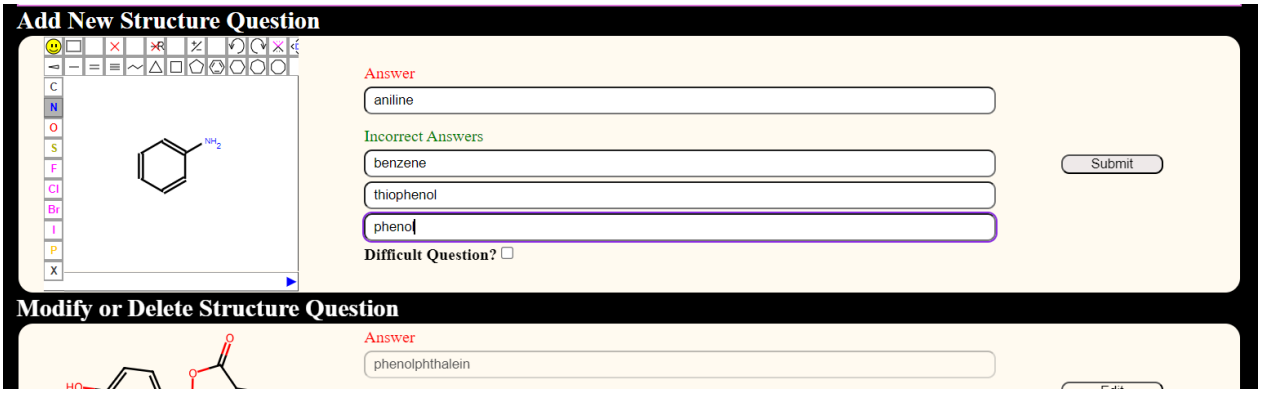


5. Admin can go to the ADMIN QUESTIONS option to

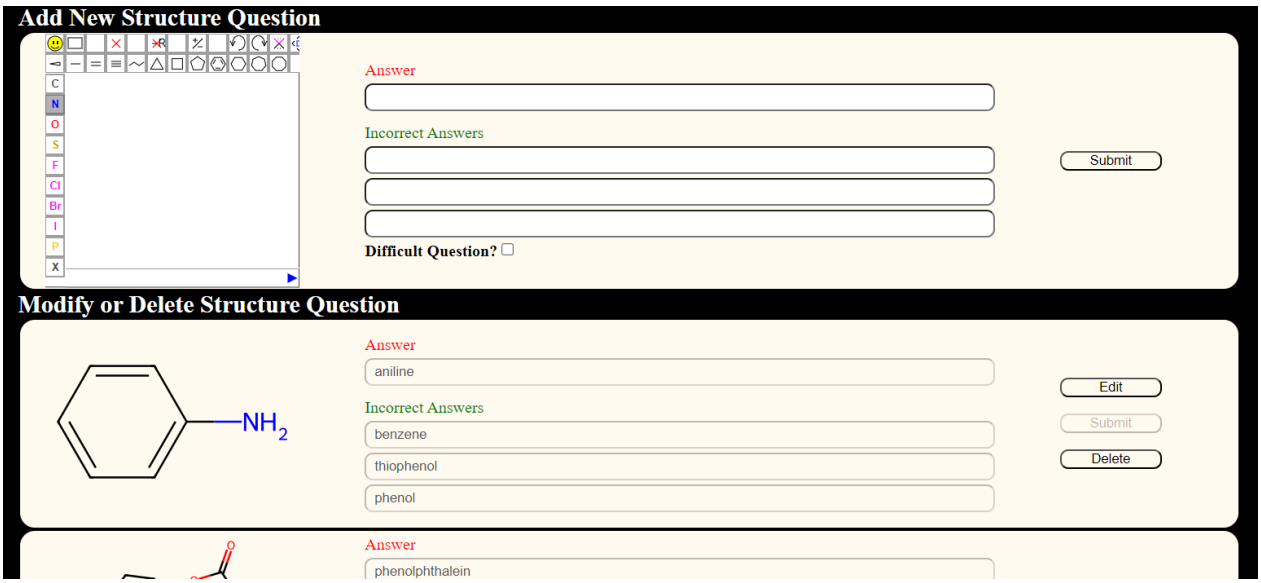
- CREATE and ADD, DELETE or MODIFY a STRUCTURE type question
- CREATE and ADD, DELETE or MODIFY a REACTION type question



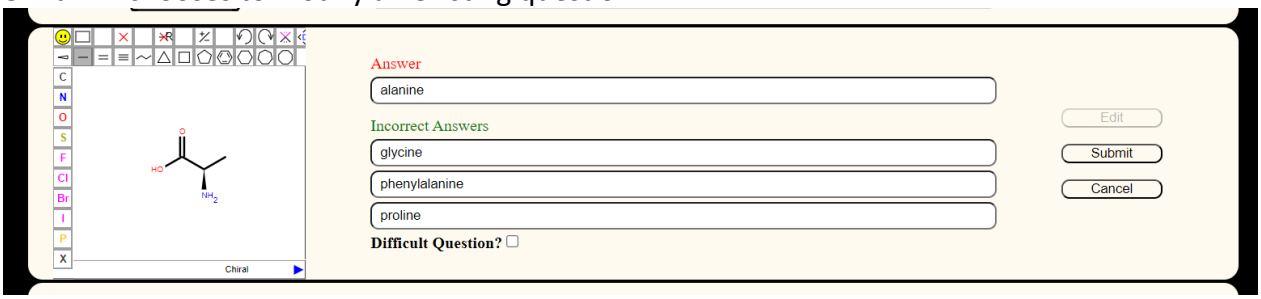
6. Admin clicks on “Structure Questions”, and chooses to create and add a new Structure-type question



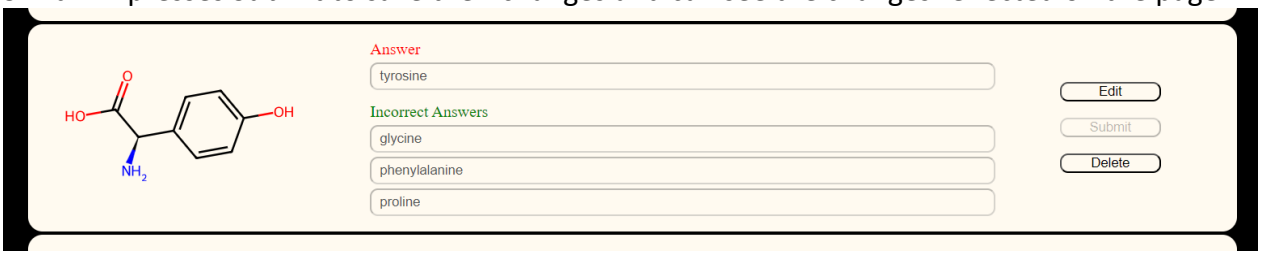
7. Admin presses submit and sees the new question in the Modify or Delete section.



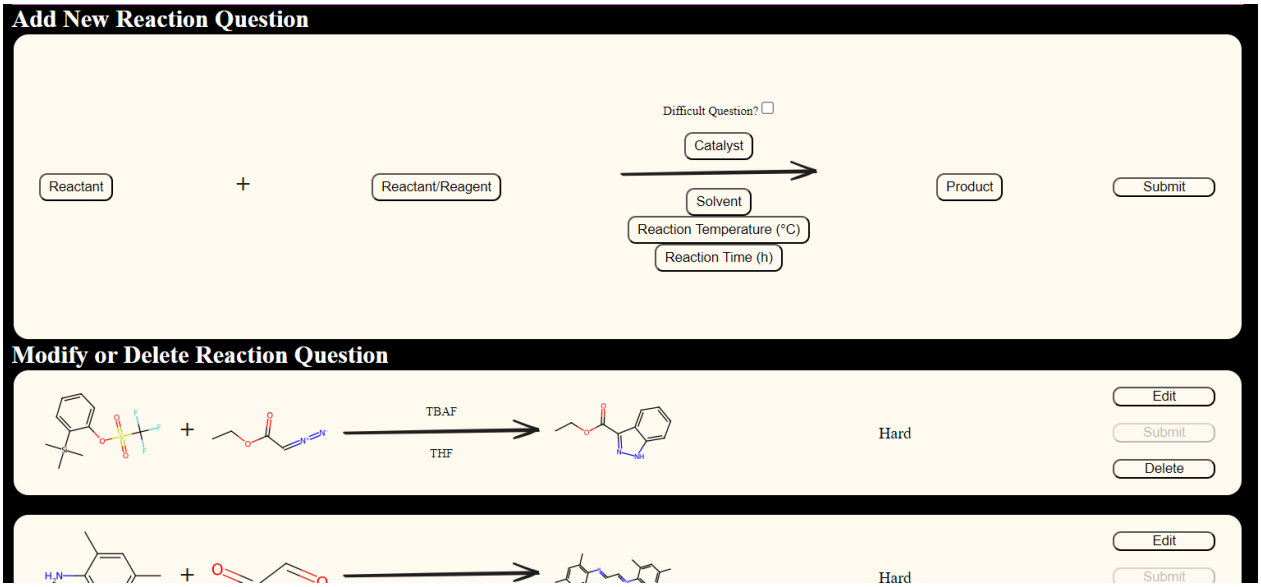
8. Admin chooses to modify an existing question.



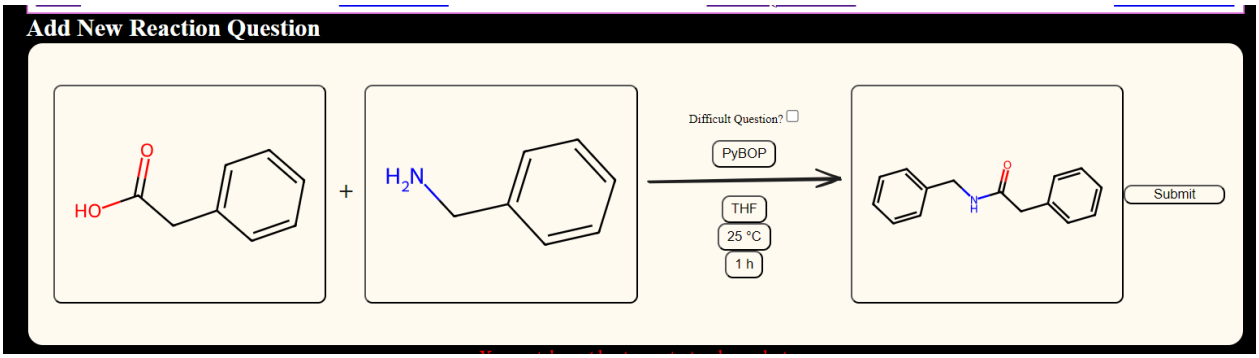
9. Admin presses Submit to save their changes and can see the changes reflected on the page.



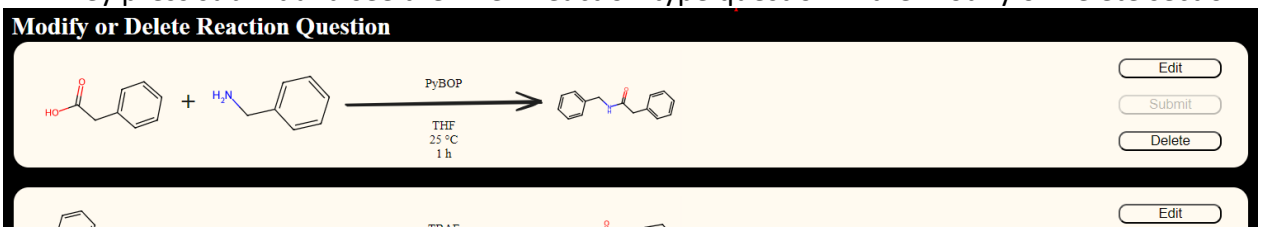
10. Admin returns to the Admin Questions menu, and selects “Reaction Questions”.



11. They add a new Reaction question by clicking on the various inputs (Reactant, Reactant/Reagent, Catalyst, Solvent, Reaction Temperature and Reaction Time) and entering the desired information.



12. They press submit and see their new reaction-type question in the Modify or Delete section.



13. Admin can go to the ‘ADMIN Questions’ option and

- MODIFY existing student scores
- DELETE existing student scores

BACK ADMIN USERS ADMIN QUESTIONS

Admin Scores Page

Scores table

id	username	score	attempt date	EDIT	DEL
5	chris	9	2024-05-07	E	X
6	chris	5	2024-05-07	E	X
7	chris	1	2024-05-07	E	X
8	chris	0	2024-05-07	E	X
9	chris	1	2024-05-07	E	X
10	chris	1	2024-05-07	E	X
11	chris	1	2024-05-07	E	X
14	chris	1	2024-05-07	E	X
15	chris	0	2024-05-07	E	X
16	chris	1	2024-05-07	E	X
17	chris	1	2024-05-07	E	X
18	chris	1	2024-05-07	E	X
19	chris	1	2024-05-07	E	X
20	chris	2	2024-05-07	E	X
21	chris	1	2024-05-07	E	X
22	chris	1	2024-05-08	E	X
23	chris	2	2024-05-08	E	X

14. Admin choose to EDIT a student score from the Scores table.

19	chris	1	2024-05-07	E	X
20	chris	2	2024-05-07	E	X
21	chris	1	2024-05-07	Cancel Save	X
22	chris	1	2024-05-08	E	X
23	chris	2	2024-05-08	E	X
24	chris	1	2024-05-08	E	X

15. Admin DELETES a student record from the table.

16	chris	1	2024-05-07	E	X
17	chris	1	2024-05-07	E	X
18	chris	1	2024-05-07	E	X
19	chris	1	2024-05-07	E	X
21	chris	1	2024-05-07	E	X
22	chris	1	2024-05-08	E	X
23	chris	2	2024-05-08	E	X
24	chris	1	2024-05-08	E	X
26	chris	1	2024-05-08	E	X

16. The record is deleted and the admin is notified of the result.

Admin Scores Page

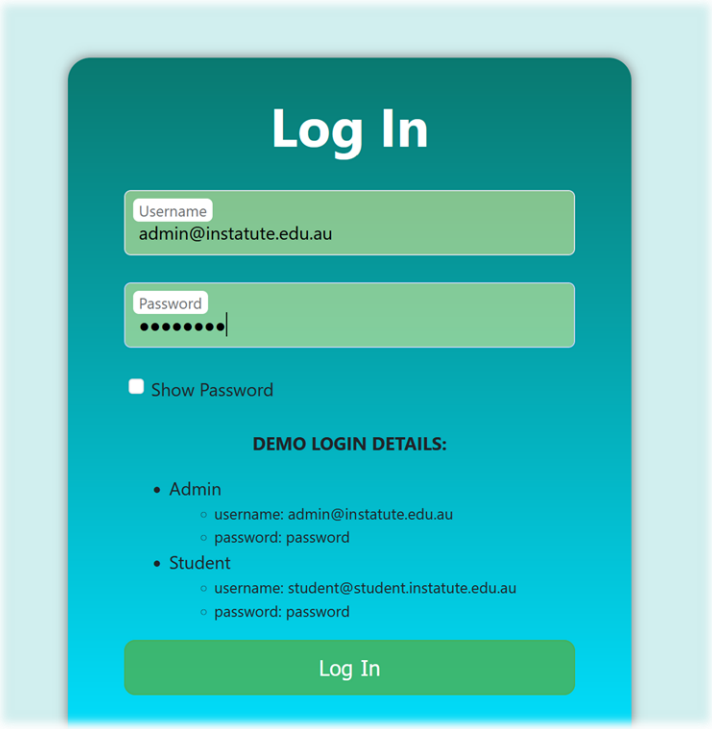
Successfully deleted the record.

Scores table

id	username	score	attempt date	EDIT	DEL
5	chris	9	2024-05-07	E	X
6	chris	5	2024-05-07	E	X
7	chris	1	2024-05-07	E	X
8	chris	0	2024-05-07	E	X
9	chris	1	2024-05-07	E	X
10	chris	1	2024-05-07	E	X
11	chris	1	2024-05-07	E	X
14	chris	1	2024-05-07	E	X
15	chris	0	2024-05-07	E	X
16	chris	1	2024-05-07	E	X

17. On pressing the BACK link in the menu.

14. On pressing the LOG IN PAGE in the menu: Admin user is logged out; the session is closed and the user is navigated to the Login page.



6.3 Abbreviations

DB: Database
InChI: International Chemical Identifier
JSON: JavaScript Object Notation
PHP: PHP Hypertext Preprocessor / Personal Home Page
SMILES: Simplified Molecular-Input Line-Entry System
SQL: Structured Query Language
SVG: Scalable Vector Graphics

6.4 List of figures and tables:

7. References

Andrew, R., Ullman, C., & Waters, C. (**2003**). Fundamental Web Design and Development Skills. *Glasshouse*.

Bienfait, B., Ertl, P. J. Cheminformatics, L. (**2013**). JSME: a free molecule editor in JavaScript. *SpringerNature*.

Delisle, M. (**2006**). Creating Your MySQL Database. *Packt Publishing Ltd*.

Full, M. (**2020**). How the Internet Works and the Web Development Process. *Independently Published*.

Gehani, N. (**2011**). The Database Application Book Using the MySQL Database System. *Apress Media, Llc*.

Landrum, G. (**2013**). RDKit.js. https://www.rdkitjs.com/

Nixon, R. (**2021**). Learning PHP, MySQL, and JavaScript. *O'Reilly Media, Inc*.

Prettyman, S. (**2020**). Learn PHP 8: using MySQL, JavaScript, CSS3, and HTML5. *Apress Media, Llc.*

Simpson, J. (**2023**). How JavaScript Works. *Apress Media Llc*.

Smith, B. (**2015**). Beginning JSON: [learn the preferred data format of the web]. *Apress Media, Llc*.

Solomons, T. W. G., Fryhle, C. B., Snyder, S. A. (**2017**) Organic chemistry, 12th ed. *John Wiley & Sons, Inc.*