The branch of “Nazarbayev Intellectual School of physics and mathematics in Uralsk” of АЕО “Nazarbayev Intellectual Schools”

Subject: Computer science

Course work

AUTOMATED SYSTEM FOR THE SUMMATIVE ASSESSMENT PROCESS

**Student:** Khabiyev Bexultan

Grade 12 “D”

**Teacher:** Zelenov Boris Alexandrovitch

Uralsk, 2022

**Content**

[Definition, investigation, and analysis 3](#_Toc93163075)

[Definition - nature of the problem 3](#_Toc93163076)

[Investigation and analysis 4](#_Toc93163077)

[Investigation 4](#_Toc93163078)

[Analysis 10](#_Toc93163079)

[Design 16](#_Toc93163080)

[Nature of the solution 16](#_Toc93163081)

[Intended benefits 37](#_Toc93163082)

[Limits of the scope solution 37](#_Toc93163083)

[Software development, programming, testing, and installation 38](#_Toc93163084)

[Development 38](#_Toc93163085)

[Programming 41](#_Toc93163086)

[Testing 66](#_Toc93163087)

[Installation 89](#_Toc93163088)

[Documentation 92](#_Toc93163089)

[User guide 92](#_Toc93163090)

[Guide to common mistakes 102](#_Toc93163091)

[Evaluation 108](#_Toc93163092)

[Discussion of the degree of success in meeting the original objectives 108](#_Toc93163093)

[Evaluate the client’s and user’s response to the system 110](#_Toc93163094)

# **Definition, investigation, and analysis**

Definition - nature of the problem  
**Description of the organization**

Nazarbayev Intellectual Schools is a network of state educational schools, where favorable conditions are created to develop creative potential and successful self-realization and integration of young citizens of Kazakhstan into society. The project for the creation of state Intellectual Schools was launched in 2008 at the initiative of the First President of the Republic of Kazakhstan, Leader of the Nation Nursultan Nazarbayev. Intellectual schools are designed to become an experimental platform for the development, observation, study, research, analysis, implementation, and realization of innovative teaching methods of modern educational programs. Since January 19, 2011, after the adoption of the Law of the Republic of Kazakhstan "On the Status of Nazarbayev University”, "Nazarbayev Intellectual Schools" and "Nazarbayev Fund", Nazarbayev Intellectual Schools have a special legal status, which provides the right to approve educational curricula independently, establish requirements for entrance examinations, current control of progress, intermediate and final certification and more.

**Description of the current system**

Schools operate based on legal acts concerning:

1) Training programs;

2) Criteria-based assessment;

3) Internal regulations for students;

4) Compliance with academic integrity;

5) The grounds for expulsion from school;

6) Living in a boarding school.

Criteria-based assessment, in turn, includes, among other things, internal summative assessment, which evaluates the speaking, listening, writing, and reading skills of the students.

In assessing listening and speaking skills, the teacher must provide students with audio material for listening and accompanying questions, after which, on the spot, evaluate each student's speech in turn. The student must prepare a speech that answers the assessment questions and deliver it to the teacher within a set time.

In assessing writing and reading skills, the teacher must provide students with a topic for writing, text, and accompanying questions, then evaluate students’ work following the established criteria. The student’s work is presented in the form of answers to the summative assessment questions.

**Description of the client activity**

My client, Aidana Khabiyeva, is a Russian language and literature teacher at Nazarbayev Intellectual School in Nur-Sultan. Aidana has been working as a teacher in the educational system of Nazarbayev Intellectual School since September of 2019. Her responsibilities include conducting lessons in the Russian language and literature and conducting a summative assessment, evaluating the listening, speaking, writing, and reading skills of students in grades 8, 9, 10. Aidana prepares summative assessment tasks for these skills by the curriculum and established criteria. At the end of the summative assessment, Aidana receives sheets with completed assignments from students.

## ****Investigation and analysis****

### Investigation

There are many methods to research the current system. However, I decided to choose observation and interview, as I think they are most suitable for my analysis.

My client and I are in the same system of Nazarbayev Intellectual Schools, but I act as a student, and my client is a teacher of the Russian language and literature. For this reason, I can directly observe and experience the work of the current system. In the course of observation, I noticed that summative assessment takes a considerable amount of time. In some cases, I had to spend additional time after lessons to pass the Speaking Assessment. Another problem is that during the summative assessment of speaking skills, the teacher may not hear some words due to extraneous noise. In such situations, the teacher does not ask the student to repeat the phrase to not interrupt the speech. This problem can become critical for the student since the teacher does not record the assessment process in any way, and he does not have the opportunity to listen to the speech again. In order to improve the current system, I asked myself the question: how can the new system reduce the time of summative assessment and store the speeches of students during the speaking.

Through interviews, I can get detailed information about the current school system from a person directly related to it. In addition, the interview will allow me to identify the main points to pay attention to when creating a new system.

**Interview**

**I: - interviewer**, C: - client

**I: - Hello Aidana, I would like to ask you a couple of questions about the current summative assessment system.**

C: - Hello, okay.

I: - **What is your profession?**

C: - I am a Russian language and literature teacher in grades 8, 9, and 10.

I: - **What information do you get from students? How do you collect this information?**

C: - I receive homework and completed assignments for summative assessments. Pupils give me their notebooks with homework for review. The completed tasks on reading and writing skills are stored on paper. In contrast, students' answers on listening and speaking skills are not stored in any way since they are evaluated immediately.

I: - **Where do you store this information?**

C: - All students' works are stored in the school, but after passing the summative assessment for a term, when I urgently need to put student grades in the electronic journal, I have to take them home.

I: - **How can software fix the current system’s shortcomings and improve it?**

C: I think the software should significantly reduce the summative assessment time to improve the current system. Students uploading audio recordings of their speaking answers will facilitate the summative assessment process. Students do not need to follow the line to pass the assessment to the teacher, as they do in school. Also, an electronic version of the current system can save me the hassle of carrying students' work home for assessment since I can access them using a computer or laptop. Even though the students' handwriting is a less significant problem for me, completing the summative assessment tasks in the Word text editor will make the students' work easier to read since they consist of printed letters.

I: - **To fix the shortcomings of the current system, I can offer you three approaches: creating a mobile application, a website, and a desktop application.**

**A mobile app is a practical and convenient option because it retains user settings and does not require all content to be downloaded. Nowadays, a vast number of people own phones, so creating a mobile application will make the new system very accessible and familiar. With the help of a mobile application, students will see the summative assessment results in a couple of taps on a smartphone, allowing the new system to be very convenient. The use of push notification in a mobile app is a very convenient way to notify students when the teacher has graded the work. Suppose your student does not have a microphone on the computer. In that case, he can make an audio recording using the phone and send it to you without sending the recording to the computer since the phone has the necessary application. Unlike other implementing methods, a mobile application can work without access to the Internet, making it less dependent on location and Internet speed. However, storing all the necessary data on a mobile device can be inconvenient if you are used to accessing it through a computer.**

**The website has several advantages over other alternative approaches. For example, the website does not require installation and therefore does not load the device’s memory on which it is used. It is essential that the website can be accessed both on a mobile phone and on a personal computer or laptop, making it less dependent than mobile and desktop applications on a specific type of device. The downside of that approach is that the website requires an internet connection, which can complicate the summative assessment process for students with low Internet speed.**

**Desktop applications are another solution. As with the mobile phone, the personal computer is a ubiquitous device found in almost every home. On the one hand, a desktop application does not require Internet access, allowing users to work in the system outside of the network coverage area. On the other hand, a desktop application binds users to a specific device (computer), making the new system less flexible. In this case, students need to carry out all the necessary actions only on a computer. In addition, desktop applications require installation.**

**I: - Which option seems best for you?**

C: - I need to access data through a computer since I spend most of my time there. Also, connecting a student with a teacher should be as simple as possible, and therefore downloading the application can create certain difficulties. For this reason, I prefer the website.

**I: - How should the new system work?**

C: - To conduct a summative assessment, I need each student to enter their name, surname, class when registering. This will help me determine whose work I am reviewing. Students will log into the system to see the summative assessment assignments that they must complete in electronic format, in a document, or using an audio recording. Once the students have completed the tasks, they should upload the work files to the website where I can download and view them. It is also crucial that students can only see their work. Thus, the new system will meet basic confidentiality requirements and prevent discrimination based on academic performance. The website must have an evaluation form to give points for the work. I want to note that I always provide feedback to my students, which helps improve their academic level, and I would like the website to have the function of leaving comments on students' work. You can add other functions for the website that you think are necessary for the new system's convenient use.

**I: - Thank you for taking your time to answer my questions.**

C: - No problem at all.

**Origin of the data**

Students' speech is not saved in any way; the assessment of listening and speaking skills is carried out on the spot immediately after the student has finished speaking. Summative writing and reading assessment are carried out on paper, where students complete assignments and answer questions. There is space on the title page of the answer sheets to fill in the student's first name, last name, and grade. After the summative assessment, these papers are collected and stored in the school office.

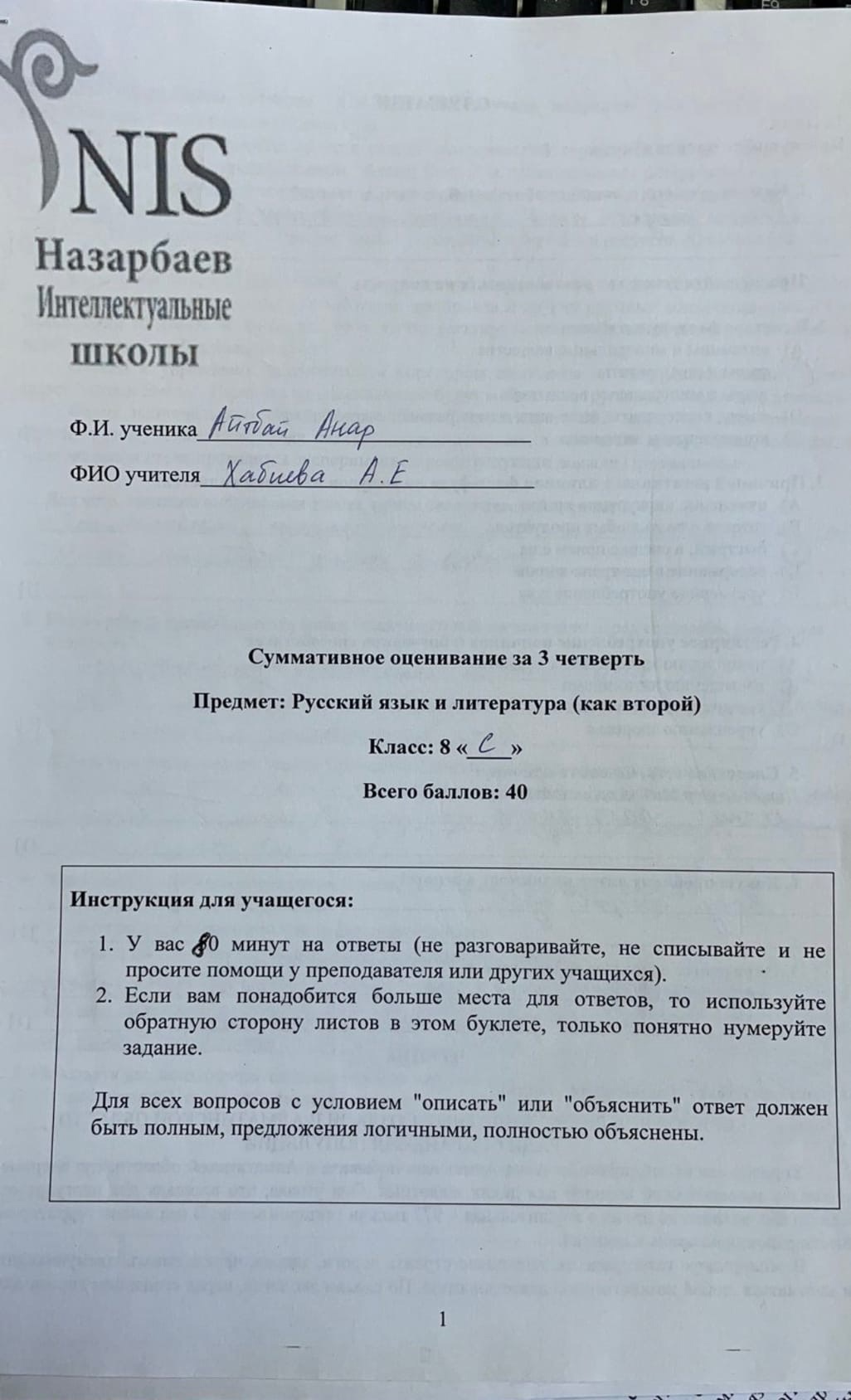


Figure 1. Title page of answer sheets

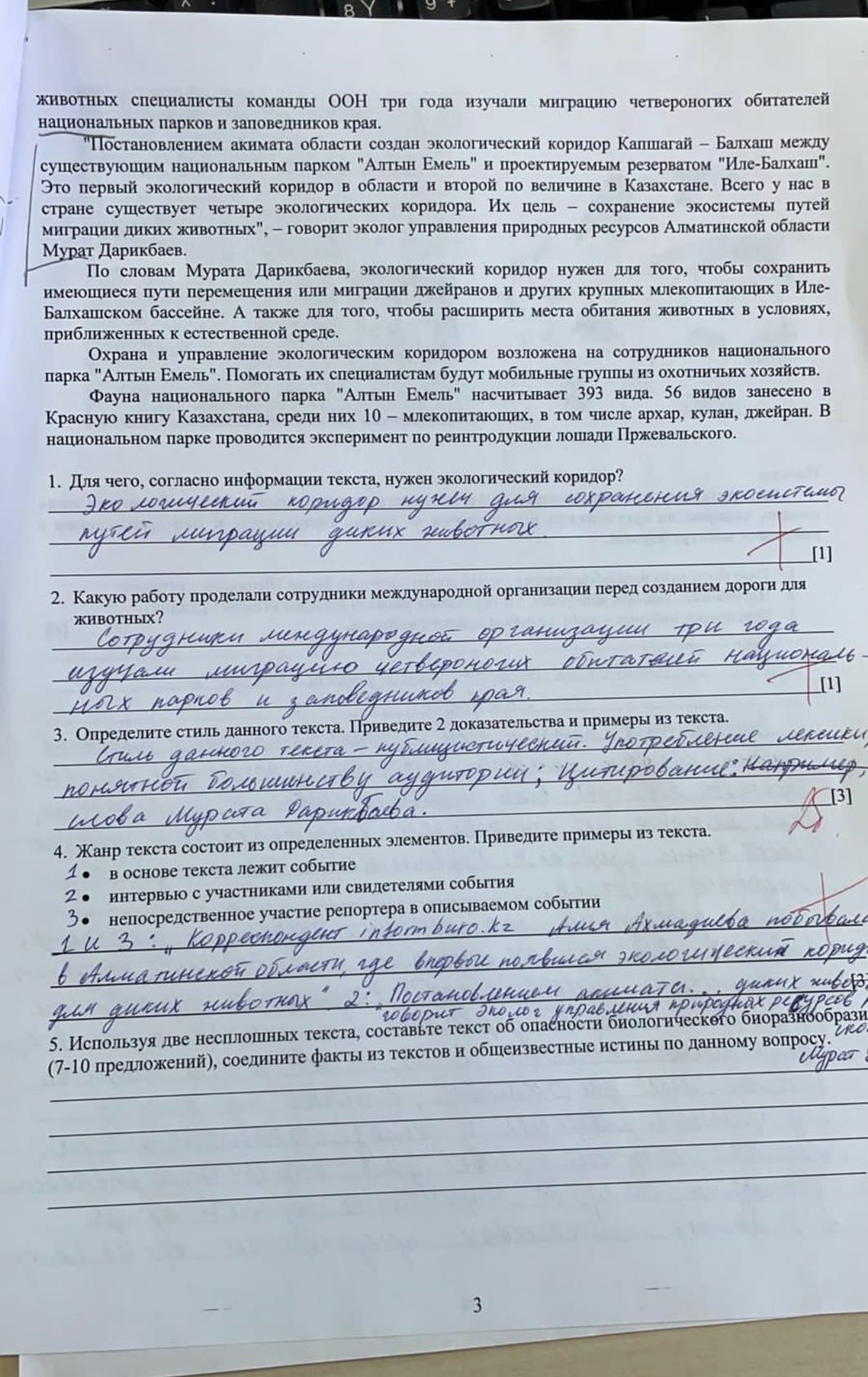


Figure 2. Student answers to summative assessment questions

**Description of the problem**

The listening and speaking assessments are time-consuming processes. Based on the observations, for the students to listen to the text twice, it will take an average of 7 minutes, taking into account that the students will be given a minute to take notes in the notebook. Then, in a set queue, which is usually represented by the alphabetical order of the first letter of the last name, the teacher must listen, evaluate and give feedback to all 20 students of the class, which takes an average of four minutes per student. Thus, the entire assessment process takes 87 minutes, roughly two 40-minute lessons.

Students' listening answers are not stored in any database, as the teacher grades speech on the spot after the student has finished speaking. This is a significant disadvantage for both the teacher and the student because if the teacher missed or did not understand a moment from the student's speech because of the background noise, he will not listen to it again. To not violate the integrity of speech and not exceed a specific time limit in which the student must go, the teacher cannot interrupt the speech and ask to repeat the unheard phrase. This may cause the teacher to give the student a lower score than he deserves.

When assessing students' reading and writing skills, the teacher may have problems perceiving information related to difficult-to-read handwriting. Just like background noise during summative speaking assessments, students' handwriting can have a negative impact on the final grade that a student receives.

### ****Analysis****

|  |  |  |
| --- | --- | --- |
| Input | Process | Output |
| Students work on summative assessment | Evaluation of work following established criteria | The final grade for the work on summative assessment |

The DFD below gives an overview of how each summative assessment usually proceeds. This process includes the student completing the assignment, evaluating the student's work, and publishing the results by the teacher.

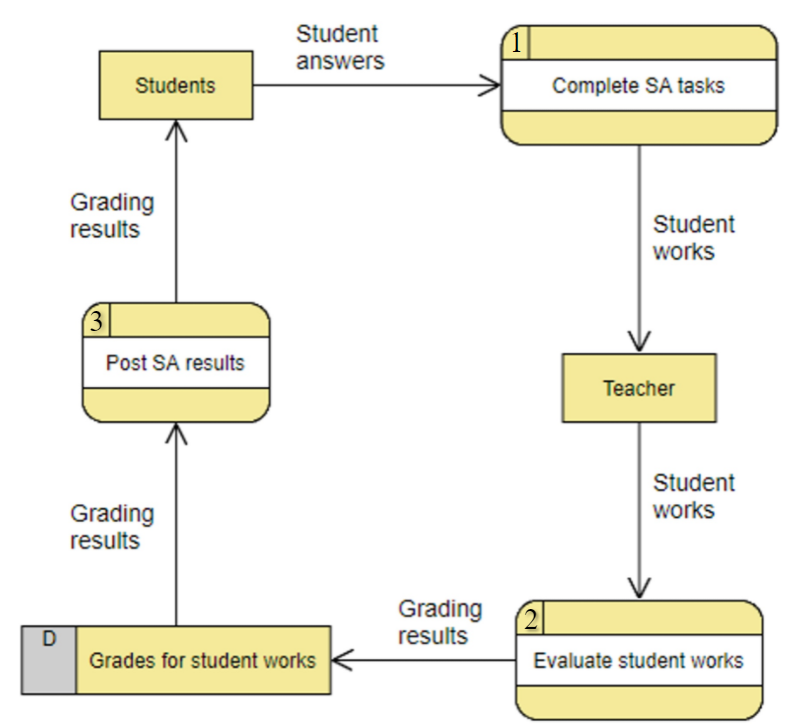


Figure 3. Data Flow Diagram of the current system

Students fill out the sheets with assignments given by the teacher during the summative assessment, answering the questions presented in them. At the end of the grading time, students return their answer sheets to the teacher.

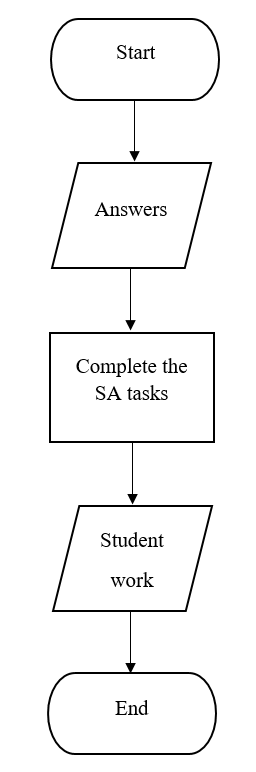


Figure 4. Process of completing a summative assessment task

After that, the teacher accepts the students' work and evaluates them on a point system following the established criteria and goals from the curriculum. The results of the students are recorded in the teacher's journal.

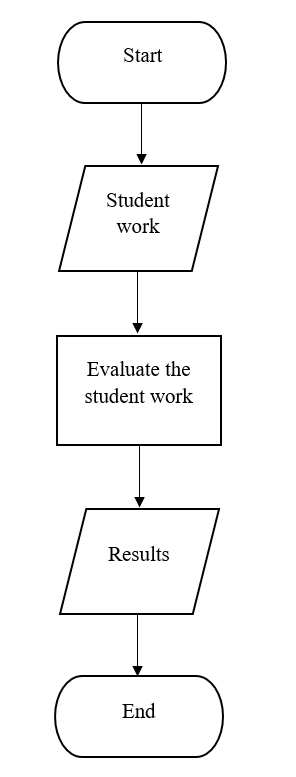
****

Figure 5. Evaluation process of student work

The teacher discusses the results with the students in the lesson and publishes the assessment results in an electronic journal.

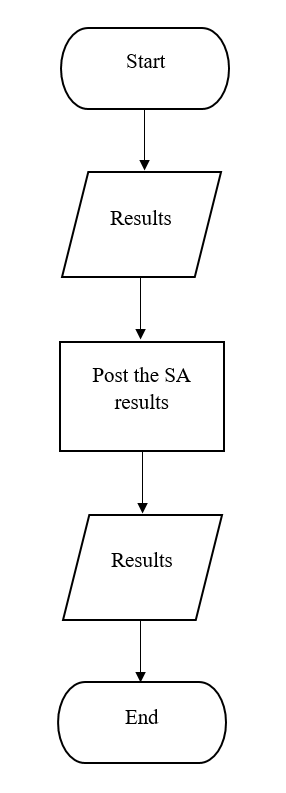
****

Figure 6. Process of publication of the assessment results

**Advantages of the current system**

Thanks to the fact that the teacher devotes separate time for each student, he can determine individual progress and understanding of the topic. A personalized approach allows the teacher to offer recommendations for improving each student’s academic level, which increases the overall level of knowledge for the entire class.

Students write their first name, last name, and grade on the title page during the summative assessment. This makes it easy to determine which student belongs to a particular answer sheet.

**Disadvantages of the current system**

The assessment process takes too much time, especially for listening and speaking skills. Students have to spend extra time to pass the summative assessment.

Student responses during the Summative Listening and Speaking Assessment are not stored in any way. Due to extraneous noise, the teacher may not hear the student, negatively affecting the grade.

Sometimes the students' handwriting makes it difficult for the teacher to read and understand the answers. This can lead to the teacher not understanding some fragments of the student's work and giving a grade that does not correspond to the student's academic level.

**Conclusion based on the analysis**

Through interviews and observations, I can conclude that the current system has significant drawbacks that negatively affect the educational process. The main disadvantages of the current system are the amount of time it takes to complete the summative assessment and the lack of data storage. The client stated that the ability to access the system through a computer and the simple connecting process is a priority, so creating a website is the best solution. In addition, Aidana noted the importance that students cannot see other people's results for summative assessment. Such a measure prevents possible bullying in school due to the poor academic performance of one of the students. Based on the study results, the current system has many areas for improvement, which can be implemented in an electronic version.

**Requirements for the new system**

1. Create a friendly and pleasant interface for the user so that he can easily navigate and have no difficulties in using the new system
2. Provide the registration of new users
3. Provide authorization to users who have already registered on the website
4. Provide the ability for all users to upload files to the new system
5. Provide an opportunity for the teacher to write feedback for students
6. Provide an opportunity for the teacher to publish the results of the summative assessments
7. Provide the ability to view uploaded files so that users can only see their works
8. Provide the ability to view the results of the assessment so that users can only see their results

A signed letter confirming the client's agreement with the new system’s requirements is presented below.



Figure 7. Letter confirming the client's agreement with the requirements of the new system

# Design

The new system should...

1. Provide registration of new users
2. Provide a friendly interface for users
3. Provide authorization of the users who are already registered in the system
4. Have a function for all users to upload files
5. Have a function for the teacher to leave comments and set results of the summative assessments
6. Provide an opportunity to see student files so that every student can only view his work
7. Provide an opportunity to see the results of the summative assessments so that every student can only view his results

The objectives presented above were made based on the requests and suggestions of the client indicated in the interview.

## Nature of the solution

**DFD**

After a detailed analysis, it was decided to create a DFD for the new advanced system. The new system has six processes: register, log in, output work, post work, evaluate work and write a comment, output results and comment. In addition, the new system has three databases that store information about users, student work, work results, and feedback attached to them. The arrows connecting the new system elements represent the direction of data flow. Thus, the DFD makes it easy to understand how the new system will work. The numbers next to the processes indicate the order in which they proceed.

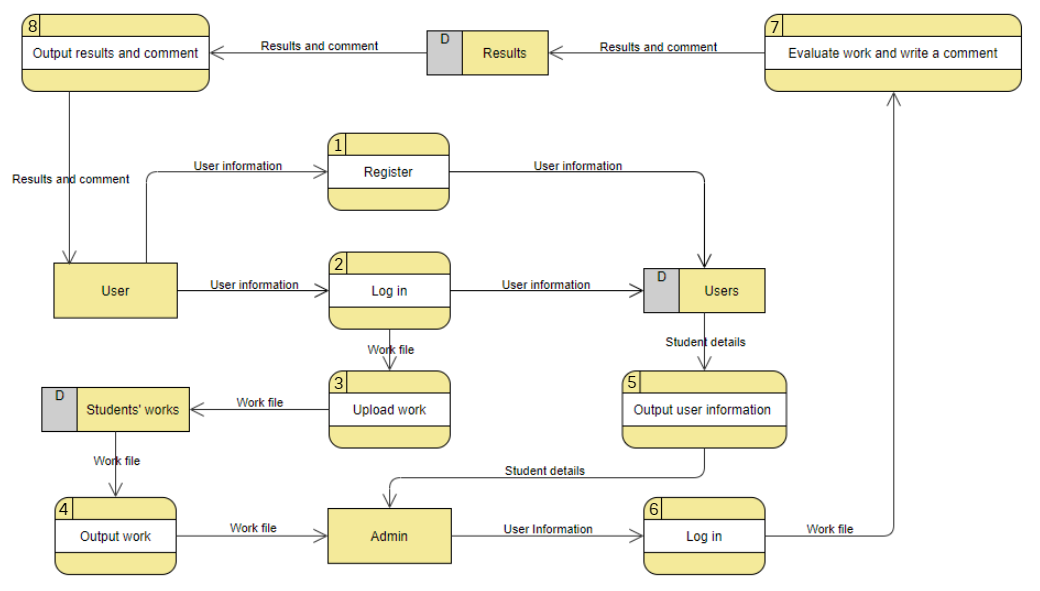


Figure 8. Data Flow Diagram of the new system

**ERD**

ERD of the new system was created to define the connections between the databases. The diagram shows that the attributes UserID, WorkID, and ResultsID are the primary keys in the tables: Users, Work, and Results, respectively. The UserID and WorkID attributes in the “Results” table and the UserID attribute in the “Work” table are also foreign keys, which provide one-to-many relationships between tables. Thus, one student can have multiple work files, but one work file cannot have multiple authors. Similarly, with the results, one student can have several results; however, several students cannot have identical results with comments.

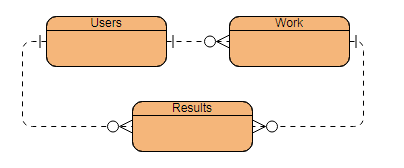


Figure 9. Entity Relationship Diagram of the new system

**Data Dictionary**

Data dictionary of table “Users”

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field name** | **Data type** | **Field Size for display** | **Validation** | **Description** | **Example** |
| UserID (PK) (AI) | int | 5 | - | Stores unique identifier of each user | 1 |
| User\_name | varchar | 30 | Length check, 2=> and <=30 characters;  Presence check | Stores first name of the user | Bexultan |
| User\_surname | varchar | 30 | Length check, 2=> and <=30 characters;  Presence check | Stores surname of the user | Khabiyev |
| User\_login | varchar | 30 | Length check, 8=> and <=30 characters;  Presence check | Stores login of the user | Khabiyev\_B |
| User\_password | varchar | 30 | Length check, 8=> and <=30 characters;  Presence check | Stores password of the user | Ilovecomputerscience |
| User\_class | varchar | 3 | Length check, <=3 characters  Presence check | Stores class of the student | 12D |
| User\_status | varchar | 10 | Presence check | Determines the status of the user, whether he is a student or a teacher | Student |

Data dictionary of table “Work”

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field name** | **Data type** | **Field Size for display** | **Validation** | **Description** | **Example** |
| WorkID (PK) (AI) | int | 5 | - | Stores unique identifier of each work | 5 |
| UserID (FK) | int | 5 | Presence check | Stores unique identifier of each user | 1 |
| Work\_name | varchar | 30 | Presence check | Stores name of the work | Russian Literature SA |
| Work\_check | int | 1 | - | Characterizes the status of the work that the student submitted. If the work has been evaluated, the field takes on the value "1"; if the work was not assessed, the field takes on the value "0". | 1 |

Data dictionary of table “Results”

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Field name** | **Data type** | **Field Size for display** | **Validation** | **Description** | **Example** |
| ResultsID (PK)(AI) | int | 5 | - | Stores unique identifier of each user | 3 |
| UserID (FK) | int | 5 | Presence check | Stores unique identifier of each user | 1 |
| WorkID (FK) | int | 5 | Presence check | Stores unique identifier of each work | 5 |
| Results\_grade | int | 3 | Length check, <=3 characters;  Presence check | Stores the number of points the student received | 12 |
| Results\_maxgrade | int | 3 | Length check, <=3 characters;  Presence check | Stores the total score the student can get for the assignment | 13 |
| Results\_comment | varchar | 250 | Length check, <=250 characters;  Presence check | Stores comment of the teacher | Your work is great |

**Flowcharts**

The flowcharts presented below demonstrate the sequence of actions in the processes of the new system.

**Registration**

To register a new user, the new system must receive the necessary personal data, including name, surname, login, password, class, and status. The user must provide all the required information, and his login must be unique to avoid further errors in the system. Otherwise, the website will issue an error notification and not complete the registration. After a successful registration of a new user, the website will display a corresponding message.

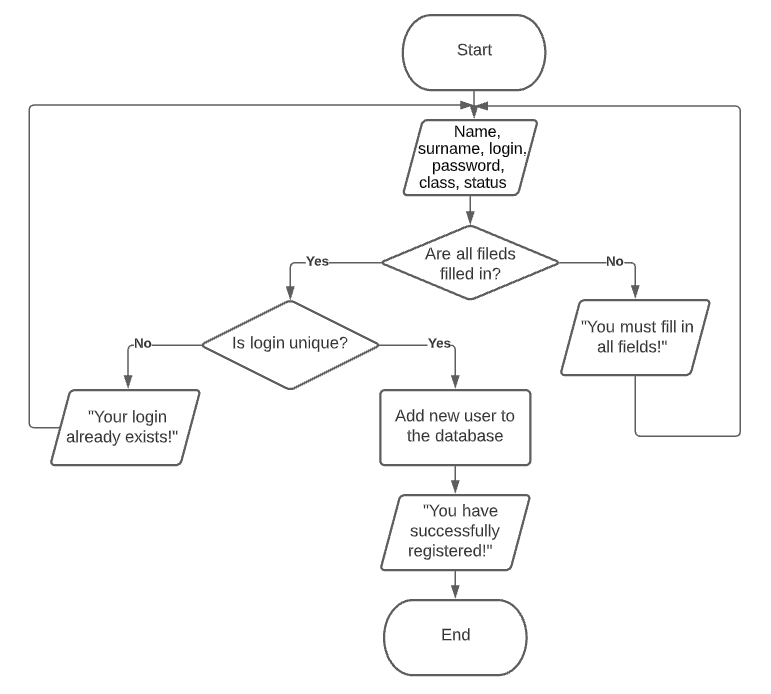


Figure 10. Registration process of the new system

**Log in**

The user must enter a login and password to log into the new system. If the entered login does not match the login of any of the users in the database, the program will give an error. Similarly, if the login does not match the password entered by the user, the program will issue an error message. After a successful authorization of a user, the website will display a corresponding notification.

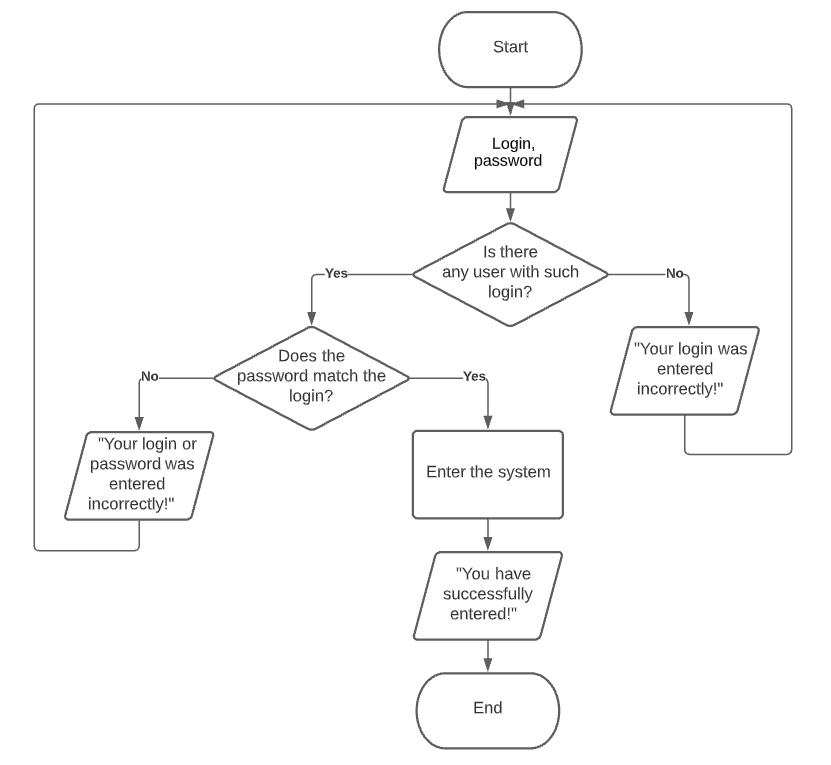


Figure 11. Authorization process of the new system

**Uploading work**

For the teacher to evaluate the student’s work, it is necessary to upload the work file to the new system. The system will notify the user whether the file has been uploaded with an appropriate message.

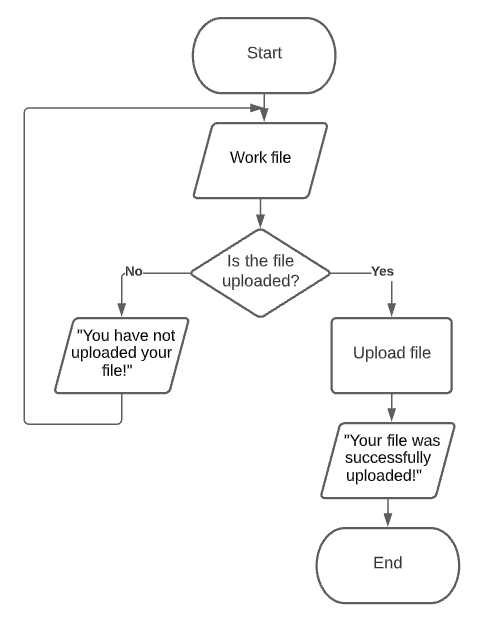


Figure 12. Process of uploading work to the new system

**Work output**

If any of the users have uploaded a work file, it will be displayed on the website

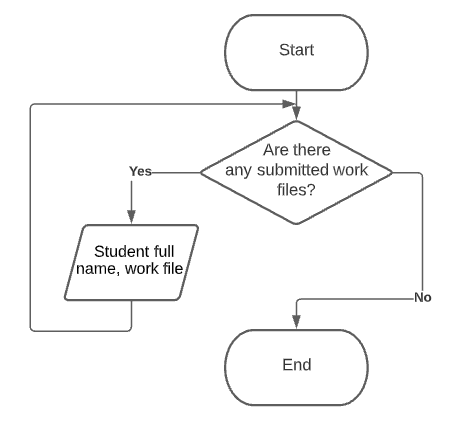


Figure 13. Work output process of the new system

**User information output**

If any of the students are registered on the website the information about them will be displayed to the admin.

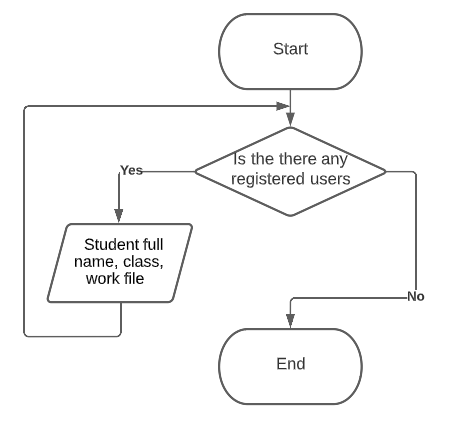


Figure 14. User information output process of the new system

**Evaluation of the student’s work**

In order for a teacher to evaluate and comment on a student's work, the teacher must enter the student's full name and select one of his work files.

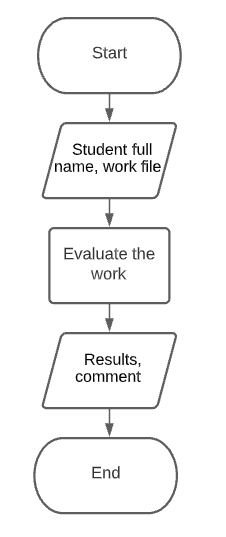


Figure 15. Evaluation process of the new system

**Results and comment output**

If the teacher has published the assessment results, they will be displayed on the website

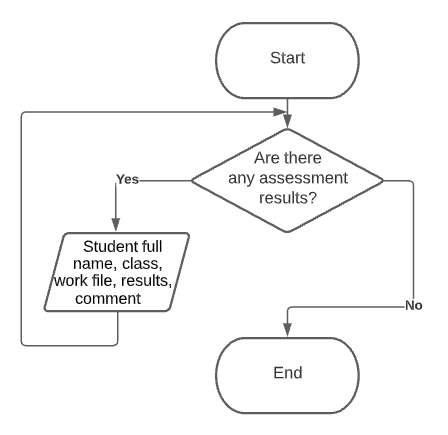


Figure 16. Results and comment output of the new system

**Menu structure**

Below is the website's menu structure, according to which the user will navigate in the new system.

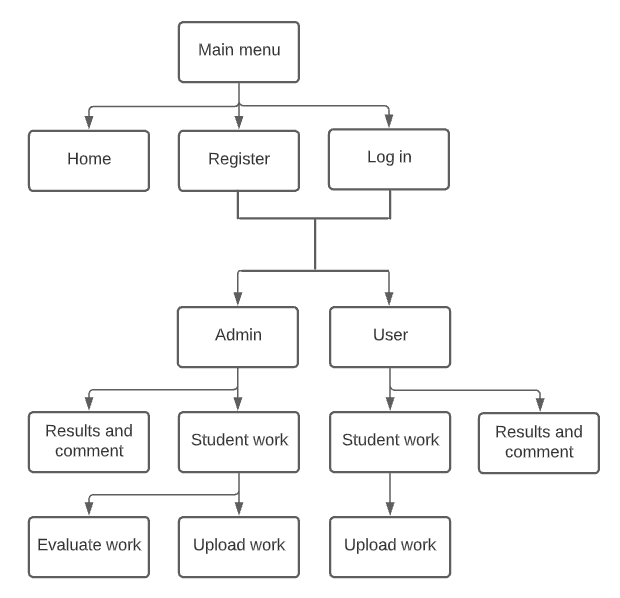


Figure 17. Menu structure of the new system

**Prototyping**

Creating a prototype of a new system before implementing a project idea is significant since it allows you to determine whether the future system will meet the client's needs and what look it will have. With the help of a prototype, it is possible to identify the shortcomings of the new system in advance**.** The prototype of the future website consists of a header, content, and footer. The header contains a navigation bar, an organization logo, and information visible on other web pages. Content is information that will be displayed differently depending on the web page.

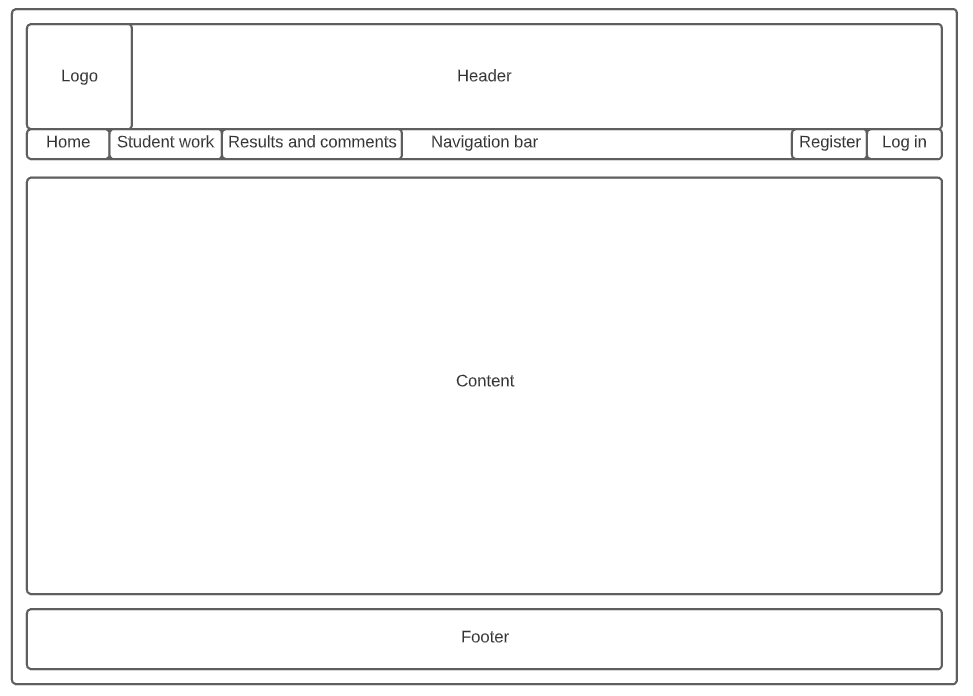
****

Figure 18. Prototype of the new system

The prototype of the new system was reviewed and approved by the client. Below is a letter confirming the client's consent.

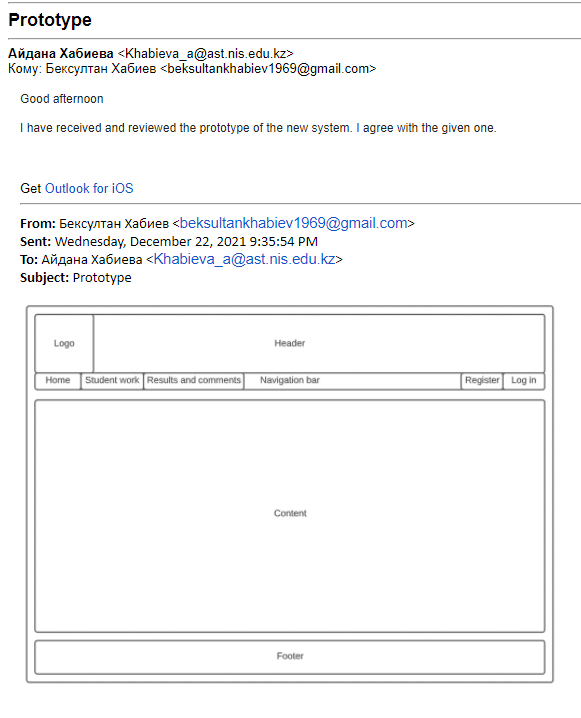


Figure 19. Letter confirming the client's agreement with the prototype of the new system

**Registration form**

This form is responsible for registering new users. The form has several fields: “Name”, “Surname”, “Login”, “Password”, “Repeat password”, “Class”, and “Status” required for filling, which include information about the user. If the user leaves any fields blank, the site issues an error notification. User login must be unique; otherwise, the user will receive an error notification. If the user repeats his password incorrectly, the website will prevent the registration of a new user. This is necessary for the user to make sure that the password he entered initially is correct and can log in the next time he visits the website.

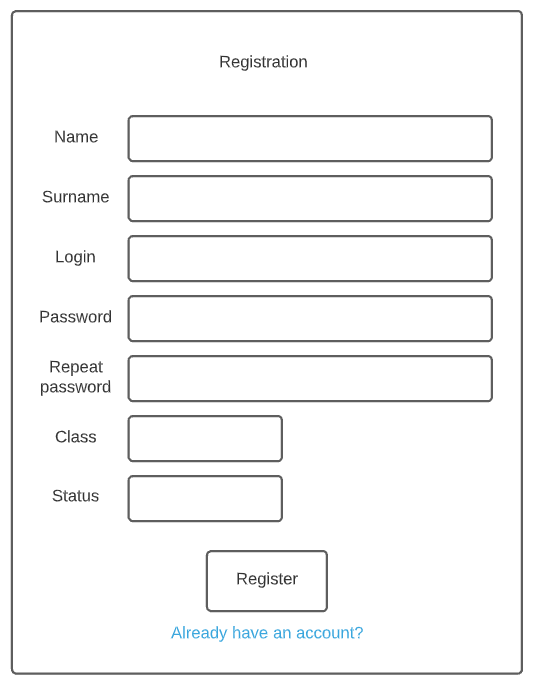


Figure 20. Registration form

**Log In form**

The presented form is responsible for authorizing users on the site. An authorized user gets full access to the functions on the site. When filling out the form, the user must write his login and password. If the user enters a non-existent login, the website will issue a notification with an error. If the user enters an existing login and password that does not match this login, he will not be able to enter the system.

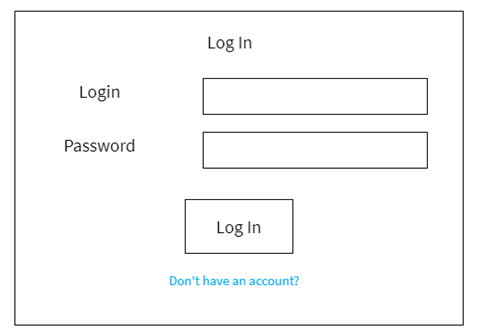


Figure 21. Log In form

**Work Upload form**

This form was created to enable students to publish their work. In this form, you can attach a file that contains the student's work and send it to the teacher.

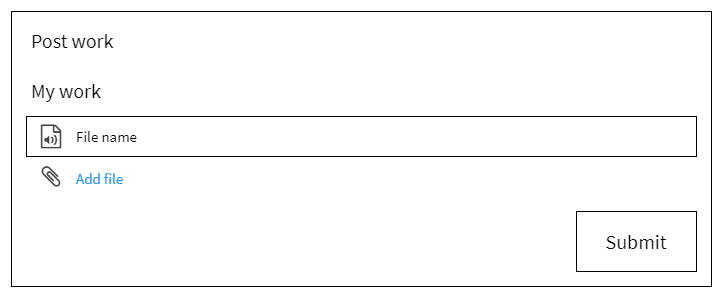


Figure 22. Work Upload form

**Output Work Form**

The following form can be seen in the "Student Work" tab. This form displays work files uploaded by students and files with assignments that the teacher uploaded, presented in the tables "Student Work" and "Assignments", respectively. Each record of these two tables contains the first name, last name, class, and work file, making it possible to determine which students completed the assignments. 

Figure 23. Work Output form

**Work Evaluation form**

This form is available only for the teacher. It allows the teacher to evaluate the work and write a commentary intended for the student.

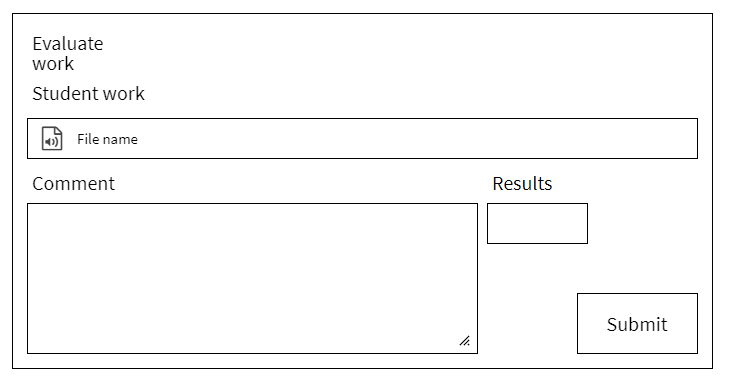


Figure 24. Work Evaluation form

**Results and Comment Output form**

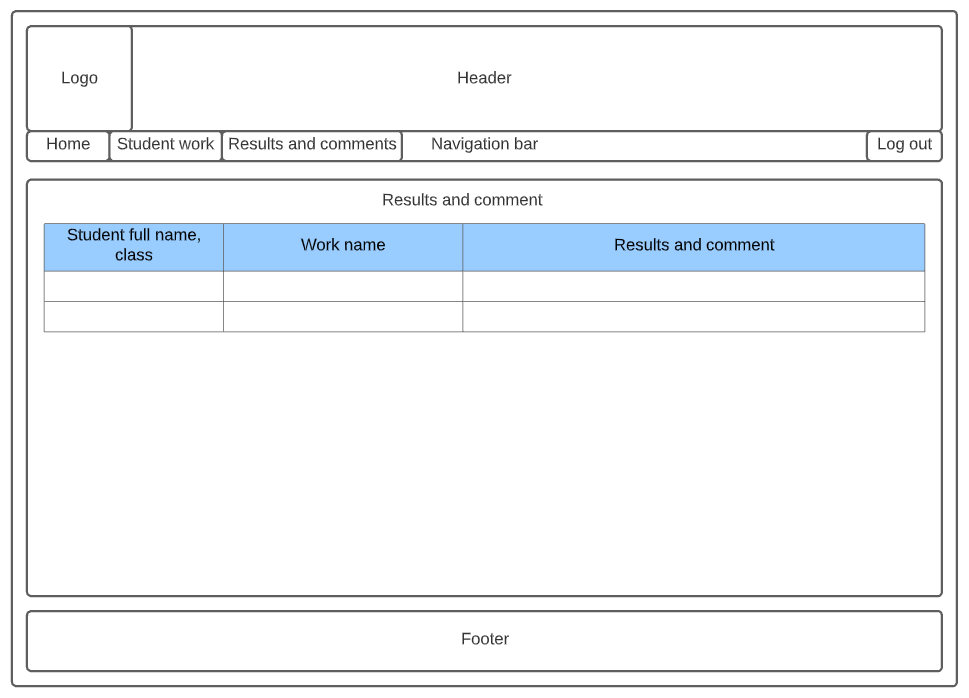
The form provided is intended for students to read the assessment results and to read the feedback that the teacher has left. All required data are presented in the "Results and comments" table. 

Figure 25. Results and Comment Output form

Hardware requirements

|  |  |  |
| --- | --- | --- |
| **Device** | **Characteristics** | **Purpose of using** |
| CPU | Intel Pentium 4 3.0GHz/ AMD Duron 2.0GHz | The processor clock speed must be high enough to make calculations in the new system |
| RAM | 4GB | The required size of RAM for stable work in Google Chrome |
| Hard disk | 60MB | Recommended hard disk space for required software installation (excluding operating system) and storing user files for one school term |
| Graphics card | AMD Radeon Xpress 1200 Series or NVIDIA GeForce FX 5200 | A graphic card is required to display the website on the screen |
| Network Interface Card | 100MB per second | A network card is required to connect the device to the Internet |
| Monitor | Screen resolution 1,280 x 800 or higher recommended  Display diagonal must be at least 15 inches | Screen resolution must meet the recommended requirements for proper display of information |
| Keyboard | Input device | Required to enter information into the database |
| Computer mouse | Control device | Needed to control the cursor while working at the computer |

Software requirements

|  |  |
| --- | --- |
| **Software** | **Purpose of using** |
| Windows | The user uses Windows as an operating system to control the work of the computer |
| Browser | An internet browser is required to access the website |
| Apache | Apache works as a local webserver |
| MySQL | Database for storing the necessary information |
| Notepad++ | Needed in order to write program code |

## Intended benefits

The presented system is more favorable for the client since it automates the assessment process, making it faster and more convenient. The new assessment format has several significant benefits over the previous one at school:

In the new system, the teacher will be able to store students' work on speaking skills. Because all student files are stored in a specific database, the teacher can re-listen to the student's speech if he has any questions. This will relieve the teacher from the difficulties of comprehending words during a live conversation and increase the accuracy of student assessment.

The new system is more time-efficient than the school system. According to the calculations indicated in the analysis, if the assessment at school takes approximately 87 minutes, then the assessment on the site will take no more than 10 minutes, taking into account the time for students to send files.

Teacher’s comments and audio files will be stored in a database, which will allow students to listen to their speech and improve it. Thus, the teacher can improve the learning process using the new system.

## Limits of the scope solution

* + - 1. Only one teacher can work on the site, so this system cannot be applied to the entire school. The reason for this is that the system was designed only for the client’s needs.
      2. To add a file to the site database, an Internet connection is required. Therefore, the time for sending students' work depends on the speed of the Internet connection.
      3. The new system does not have a webcam recording function to simulate the conditions during a live conversation since many students do not have webcams, and such videos can take up a lot of space and can be challenging to play.

# Software development, programming, testing, and installation

## Development

The diagram below shows the arrangement of the future system. The names of the forms and tables presented in the schema characterize the processes in the forms or the data stored in the tables. Since my client has approved the addition of other features, I have created several essential functions to provide more functionality for users, such as “Edit work”, “Edit results” and “Delete results”.

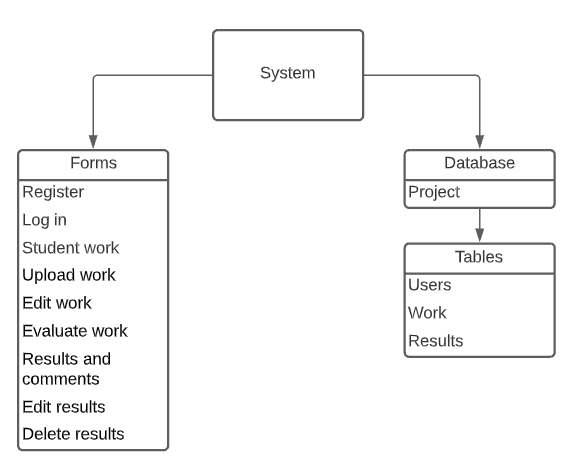


Figure 26. Structure of the new system

**ERD**

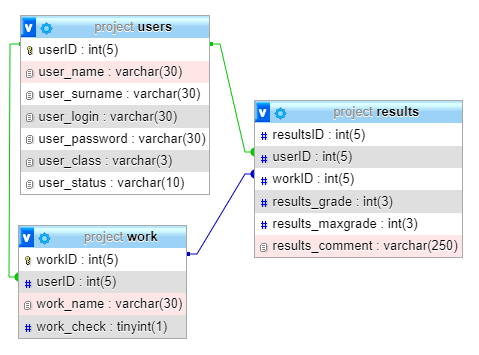
****

Figure 27. Entity Relationship Diagram of the new system

The following are the structures of tables that store system data. The table’s structure allows you to see the data types, the maximum number of characters, and the presence of the primary and foreign keys used in the system.

Table “Users”

The "Users" table stores personal data of users that was obtained during the registration process

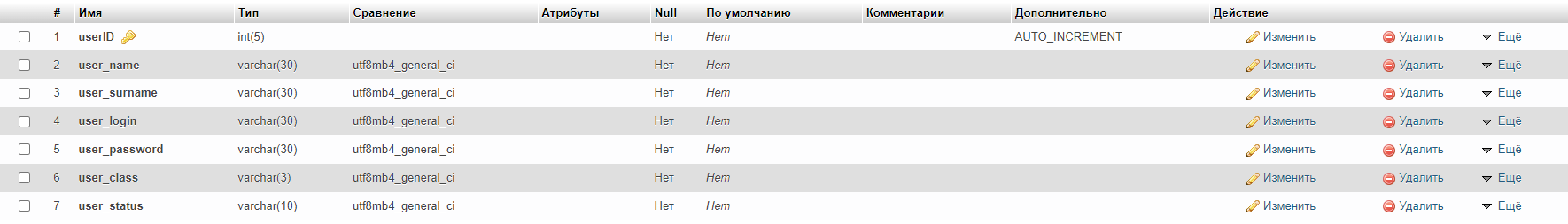


Figure 28. Users table

Table “Work”

The "Work" table stores student files that contain completed assignments.

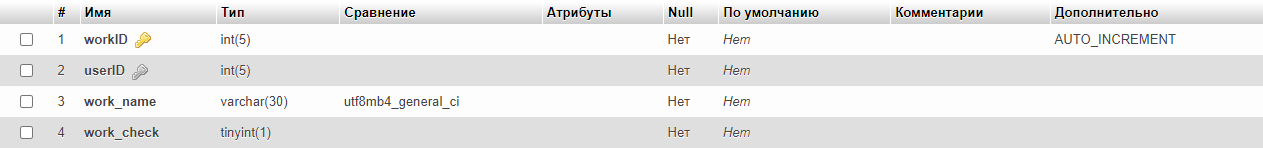


Figure 29. Work table

Table “Results”

The "Results" table stores the results and teacher comments on student work.

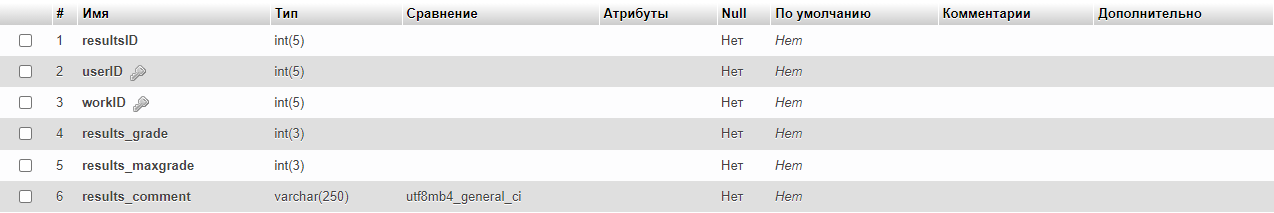


Figure 30. Results table

## Programming

**Connection with database**

File “connect.php”

<?php

$servername = "localhost";

$name1 = "root";

$password = "";

$database = "project";

*// This line establishes connection with the database*

$conn = mysqli\_connect($servername, $name1, $password, $database);

*// If the connection is not established, then the program will give an error*

if (!$conn)

{

die("Connection failed: " . mysqli\_connect\_error());

}

?>

**Registration form**

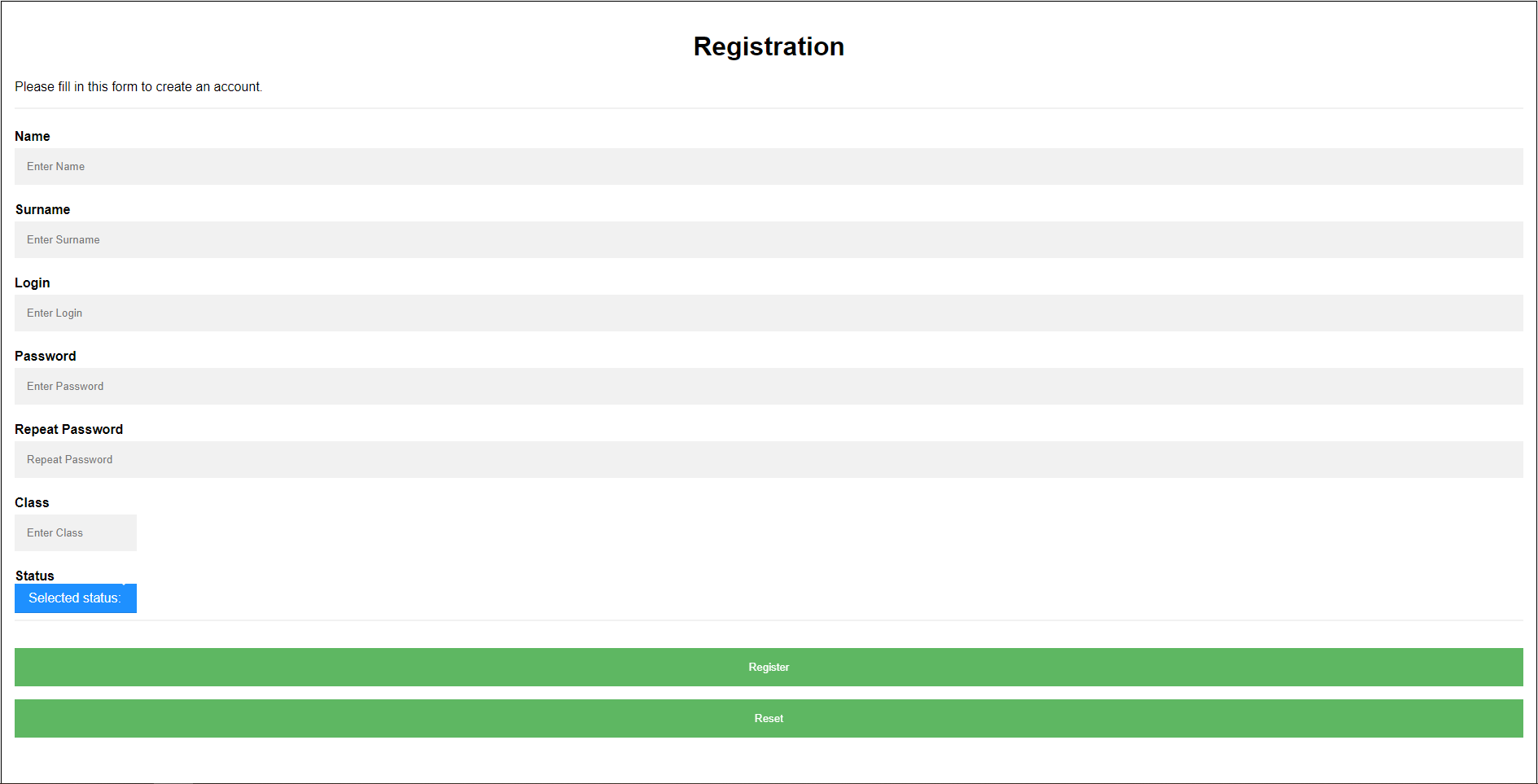
****

Figure 31. Registration form

File “signup.php”

<form action ="handler.php" method ="POST">

<div class="container">

<h1 align="center">Registration</h1>

<p>Please fill in this form to create an account.</p>

<hr>

<label for="name"><b>Name</b></label>

<input type="text" placeholder="Enter Name" name="name" required>

<label for="surname"><b>Surname</b></label>

<input type="text" placeholder="Enter Surname" name="surname" required>

<label for="login"><b>Login</b></label>

<input type="text" placeholder="Enter Login" name="login" required>

<label for="password"><b>Password</b></label>

<input type="password" placeholder="Enter Password" name="password" id="psw" required>

<label for="psw-repeat"><b>Repeat Password</b></label>

<input type="password" placeholder="Repeat Password" name="psw-repeat" id="psw-repeat" required>

<label for="class"><b>Class</b></label>

<br>

<input type="text" style="width: 150px;" placeholder="Enter Class" name="class" id="class">

<div class="custom-select" style="width: 150px;">

<label for="status"><b>Status</b></label>

<select name="status">

<option value="0">Selected status:</option>

<option value="student">Student</option>

<option value="teacher">Teacher</option>

</select>

</div>

<hr>

<button type="submit" class="registerbtn">Register</button>

<button type="reset" class="registerbtn">Reset</button>

</div>

<div class="container signin">

<p>Already have an account? <a href="login.php">Log in</a>.</p>

</div>

</form>

File “handler.php”

<?php

*// Creating a session to store data that will be used on several pages of the site*

session\_start();

*// Transfer data from the form and define variables for it*

$user\_name = $\_POST["name"];

$user\_surname = $\_POST["surname"];

$user\_login = $\_POST["login"];

$user\_password = $\_POST["password"];

$user\_class = $\_POST["class"];

$user\_status = $\_POST["status"];

$psw\_repeat = $\_POST["psw-repeat"];

*// This condition check for the presence of data. If the user left the input field blank, the program will generate an error*

if (empty($user\_status))

{

print '<div class="alert alert-danger"><strong>You must enter your status!</strong> You should <a href="signup.php" class="alert-link">try again</a>.</div>';

exit();

}

*// These conditions check the length of the entered data. If the number of characters does not fit in the allowed interval, then the program will generate an error*

elseif (strlen($user\_name) > 30 OR strlen($user\_name) < 2)

{

print '<div class="alert alert-danger"><strong>Your name is longer than 30 characters or shorter than 2 characters!</strong> You should <a href="signup.php" class="alert-link">try again</a>.</div>';

exit();

}

elseif (strlen($user\_surname) > 30 OR strlen($user\_surname) < 2)

{

print '<div class="alert alert-danger"><strong>Your surname is longer than 30 characters or shorter than 2 characters!</strong> You should <a href="signup.php" class="alert-link">try again</a>.</div>';

exit();

}

elseif (strlen($user\_login) > 30 OR strlen($user\_login) < 8)

{

print '<div class="alert alert-danger"><strong>Your login is longer than 30 characters or shorter than 8 characters!</strong> You should <a href="signup.php" class="alert-link">try again</a>.</div>';

exit();

}

elseif (strlen($user\_password) > 30 OR strlen($user\_password) < 8)

{

print '<div class="alert alert-danger"><strong>Your password is longer than 30 characters or shorter than 8 characters!</strong> You should <a href="signup.php" class="alert-link">try again</a>.</div>';

exit();

}

elseif ($user\_status == 'student')

{

if (!empty($user\_class))

{

if (strlen($user\_class) > 3)

{

print '<div class="alert alert-danger"><strong>Your class name is longer than 3 characters!</strong> You should <a href="signup.php" class="alert-link">try again</a>.</div>';

exit();

}

}

else

{

print '<div class="alert alert-danger"><strong>You must enter your class name!</strong> You should <a href="signup.php" class="alert-link">try again</a>.</div>';

exit();

}

}

*// Database connection*

include ("connect.php");

*// Presented code searches in database for an ID whose login matches the login from the form*

$searchlogin = mysqli\_query($conn, "SELECT userID FROM users WHERE user\_login = '$user\_login'");

$myrow = mysqli\_fetch\_array($searchlogin);

*// If there is a user with the same login, then the program will give an error. This is necessary to check the uniqueness of the login*

if (!empty($myrow['userID']))

{

print '<div class="alert alert-danger"><strong>Your login is already exists!</strong> You should <a href="signup.php" class="alert-link">try again</a>.</div>';

exit();

}

*// If the user incorrectly repeated his own password, the program will generate an error. This condition eliminates the possibility that the user's password was initially entered incorrectly*

elseif ($user\_password !== $psw\_repeat)

{

print '<div class="alert alert-danger"><strong>Passwords do not match!</strong> You should <a href="signup.php" class="alert-link">try again</a>.</div>';

exit();

}

// This line encrypts the password entered by the user so that no one else can use it, thereby increasing the security of the system

$hash\_password = md5($user\_password);

*// Insertion of the data entered by the user into one of the database tables*

$add = "INSERT INTO users (user\_name, user\_surname, user\_login, user\_password, user\_class, user\_status) VALUES ('$user\_name', '$user\_surname', '$user\_login', '$hash\_password ', '$user\_class', '$user\_status')";

*// This condition checks if the user data was inserted successfully*

if (mysqli\_query($conn, $add))

{

*// We select a record from the database in which the login is equal to the login entered by the user*

$query = mysqli\_query($conn,"SELECT \* FROM users WHERE user\_login ='$user\_login'");

$data = mysqli\_fetch\_assoc($query);

*// This array is needed to store user data while he is on the site*

$\_SESSION['data'] =

[

"userID" => $data['userID'],

"user\_name" => $data['user\_name'],

"user\_surname" => $data['user\_surname'],

"user\_login" => $data['user\_login'],

"user\_class" => $data['user\_class'],

"user\_status" => $data['user\_status']

];

print '<div class="alert alert-success"><strong>You have successfully registered!</strong> Go to <a href="index.php" class="alert-link">home page</a>.</div>';

}

else

{

print '<div class="alert alert-danger"><strong>Error!</strong> You should <a href="signup.php" class="alert-link">try again</a>.</div>';

exit();

}

?>

**The “Log in” form**

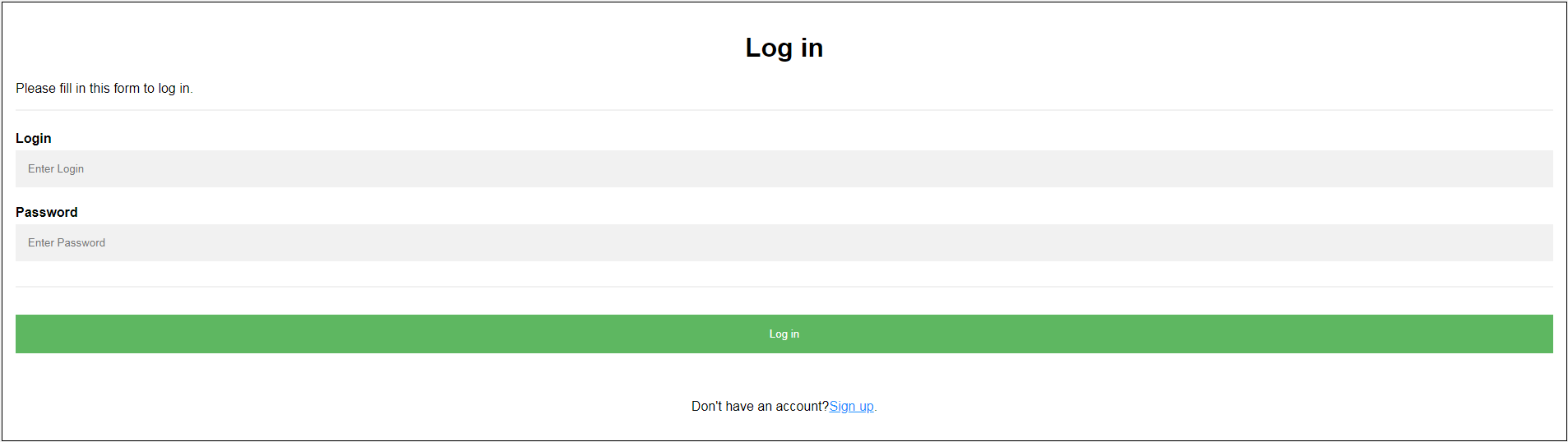


Figure 32. Log In form

File “login.php”

<form action ="handler1.php" method ="POST">

<div class="container">

<h1 align="center">Log in</h1>

<p>Please fill in this form to log in.</p>

<hr>

<label for="login"><b>Login</b></label>

<input type="text" placeholder="Enter Login" name="login" required>

<label for="password"><b>Password</b></label>

<input type="password" placeholder="Enter Password" name="password" id="psw" required>

<hr>

<button type="submit" class="loginbtn" name="submit">Log in</button>

</div>

<div class="container signup">

<p>Don't have an account?<a href="signup.php">Sign up</a>.</p>

</div>

</form>

File “handler1.php”

<?php

*// Creating a session to store data that will be used on several pages of the site*

session\_start();

*// Database connection*

include('connect.php');

if (isset($\_POST['submit']))

{

*// Transfer data from the form and define variables for it*

$user\_login = $\_POST['login'];

$user\_password = $\_POST['password'];

*// We select a record from the database in which the login is equal to the login entered by the user*

$query = mysqli\_query($conn,"SELECT \* FROM users WHERE user\_login='$user\_login'");

$data = mysqli\_fetch\_assoc($query);

//

if (!empty($data['userID']))

{

*// This array is needed to store user data while he is on the site*

$\_SESSION['data'] =

[

"userID" => $data['userID'],

"user\_name" => $data['user\_name'],

"user\_surname" => $data['user\_surname'],

"user\_login" => $data['user\_login'],

"user\_status" => $data['user\_status'],

];

*// The condition compares the password from the database and the password entered by the user. If the passwords do not match, then the programs will give an error*

if ($data['user\_password'] === ($\_POST['password']))

{

print '<div class="alert alert-success"><strong>You have successfully entered!</strong> Go to <a href="index.php" class="alert-link">home page</a>.</div>';

}

else

{

print '<div class="alert alert-danger"><strong>Your login or password was entered incorectly!</strong> You should <a href="login.php" class="alert-link">try again</a>.</div>';

exit();

}

}

else

{

print '<div class="alert alert-danger"><strong>Your login or password was entered incorectly!</strong> You should <a href="login.php" class="alert-link">try again</a>.</div>';

exit();

}

}

?>

**The “Post work” form**



File “work.php”

<form enctype="multipart/form-data" action="handler2.php" method="POST">

<p>Custom file:</p>

<div class="custom-file mb-3">

<input type="file" class="custom-file-input" name="file">

<label class="custom-file-label" for="file">Choose file</label>

</div>

<div class="mt-3">

<button type="submit" class="btn btn-primary" name="submit">Submit</button>

</div>

</form>

File “handler2.php”

<?php

*// Creating a session to store data that will be used on several pages of the site*

session\_start();

*// Transfer data from the form and define variables for it*

**$userID = $\_SESSION['data']['userID'];**

$name = $\_FILES['file']['name'];

$tmp\_name = $\_FILES['file']['tmp\_name'];

$submitbutton = $\_POST['submit'];

$position = strpos($name, ".");

$fileextension = substr($name, $position + 1);

$fileextension = strtolower($fileextension);

*// If a file name exists, then the path is set for it*

if (isset($name))

{

$path= 'files/';

if (!empty($name))

{

*// Move the downloaded file to the location specified in the variable $path. If the file has been moved, a corresponding notification will be issued*

if (move\_uploaded\_file($tmp\_name, $path.$name))

{

print '<div class="alert alert-success"><strong>Your file was successfully uploaded!</strong> Go back to <a href="main1.php" class="alert-link">home page</a>.</div>';

}

else

{

print '<div class="alert alert-danger"><strong>Error!</strong> You should <a href="work.php" class="alert-link">try again</a>.</div>';

exit();

}

}

else

{

print '<div class="alert alert-danger"><strong>You have not uploaded your file!</strong> You should <a href="work.php" class="alert-link">try again</a>.</div>';

exit();

}

}

*// Database connection*

include('connect.php');

*// Insertion of the data entered by the user into one of the database tables*

$myrow = mysqli\_query($conn, “INSERT INTO work (userID, work\_name) VALUES (‘$userID’, ‘$name’)”);

mysqli\_close($conn);

?>

**The “Edit work” form**

****

Figure 33. Work Editing form

File “edit.php”

<form enctype="multipart/form-data" action="edithandler.php" method="POST">

<div class="form-group" style="width: 10%;">

<label for="workID">WorkID:</label>

<input type="number" class="form-control" name="workID" required>

</div>

<p>Custom file:</p>

<div class="custom-file mb-3">

<input type="file" class="custom-file-input" name="file">

<label class="custom-file-label" for="file">Choose file</label>

</div>

<div class="mt-3">

<button type="submit" class="btn btn-primary" name="submit">Submit</button>

</div>

</form>

File “edithandler.php”

<?php

*// Database connection*

include('connect.php');

*// Creating a session to store data that will be used on several pages of the site*

session\_start();

*// Transfer data from the form and define variables for it*

$userID = $\_SESSION['data']['userID'];

$workID = $\_POST['workID'];

$name = $\_FILES['file']['name'];

$tmp\_name = $\_FILES['file']['tmp\_name'];

$submitbutton = $\_POST['submit'];

$position = strpos($name, ".");

$fileextension = substr($name, $position + 1);

$fileextension = strtolower($fileextension);

$add = mysqli\_query($conn, "SELECT \* FROM work WHERE userID = '$userID' AND work\_check ='0'");

while ($row = mysqli\_fetch\_array($add))

{

$ar[] = $row['workID'];

}

*// This condition checks if the work that the user wants to edit belongs to the same user. If the user enters the ID of a work that was not uploaded by him, the program will notify him of this*

if (in\_array($workID, $ar, FALSE))

{

*// If a file name exists, then the path is set for it*

$path= 'files/';

if (!empty($name))

{

*// Move the downloaded file to the location specified in the variable $path. If the file has been moved, a corresponding notification will be issued*

if (move\_uploaded\_file($tmp\_name, $path.$name))

{

print '<div class="alert alert-success"><strong>Your file was successfully uploaded!</strong> Go back to <a href="main1.php" class="alert-link">home page</a>.</div>';

}

else

{

print '<div class="alert alert-danger"><strong>Error!</strong> You should <a href="edit.php" class="alert-link">try again</a>.</div>';

exit();

}

}

else

{

print '<div class="alert alert-danger"><strong>You have not uploaded your file!</strong> You should <a href="edit.php" class="alert-link">try again</a>.</div>';

exit();

}

}

else

{

print '<div class="alert alert-danger"><strong>You are trying to edit work of another student or that does not exist!</strong> You should <a href="edit.php" class="alert-link">try again</a>.</div>';

exit();

}

*// Insertion of the data entered by the user into one of the database tables*

$myrow = mysqli\_query($conn, "UPDATE work SET work\_name = '$name' WHERE workID = '$workID'");

mysqli\_close($conn);

?>

**The “Evaluate work” form**

****

Figure 34. Work evaluation form

File “evaluate.php”

<form action="handler3.php" method="POST">

<table>

<tr>

<td>

<div class="form-group" style="width: 90%;">

<label for="workID">WorkID:</label>

<input type="number" class="form-control" name="workID" required>

</div>

</td>

<td>

<div class="form-group">

<label for="fullname">Select student:</label>

<select name="fullname" class="form-control">

<?php

include('connect.php');

$myrow = mysqli\_query($conn, "SELECT \* FROM users WHERE user\_status ='student'");

while ($row = mysqli\_fetch\_array($myrow))

{

echo "<option>". $row['user\_surname'], ' ', $row['user\_name'] ."</option>";

}

?>

</select>

</div>

</td>

</tr>

</table>

<table>

<tr>

<td>

<div class="form-group" style="width: 90%;">

<label for="grade">Number of points:</label>

<input type="number" class="form-control" name="grade" required>

</div>

</td>

<td>

<div class="form-group" style="width: 90%;">

<label for="maxgrade">Total score:</label>

<input type="number" class="form-control" name="maxgrade" required>

</div>

</td>

</tr>

</table>

<div class="form-group" style="width: 50%;">

<label for="comment">Comment:</label>

<textarea class="form-control" rows="5" name="comment" required></textarea>

</div>

<div class="mt-3">

<button type="submit" class="btn btn-primary">Submit</button>

</div>

</form>

File “handler3.php”

<?php

*// Database connection*

include (“connect.php”);

*// Transfer data from the form and define variables for it*

$user\_fullname = $\_POST[“fullname”];

*// This SQL statement looks for users with a full name similar to the full name obtained from the form*

$searchfullname = mysqli\_query($conn, “SELECT userID FROM users WHERE ‘$user\_fullname’ = CONCAT(user\_surname, ‘ ‘, user\_name)”);

$myrow = mysqli\_fetch\_array($searchfullname);

*// If there is a user ID, where his full name matches the full name entered by the user, then the program will continue executing the algorithm*

if (!empty($myrow[‘userID’]))

{

*// Transfer data from the form and define variables for it*

$userID = $myrow[“userID”];

$workID = $\_POST[“workID”];

$results\_grade = $\_POST[“grade”];

$results\_maxgrade = $\_POST[“maxgrade”];

$results\_comment = $\_POST[“comment”];

*// These conditions check the length of the entered data. If the number of characters doesn't fill in the allowed range, then the program will generate an error*

if (strlen($results\_grade) > 3)

{

print '<div class="alert alert-danger"><strong>Your number of points is more than the three-digit number!</strong> You should <a href="evaluate.php" class="alert-link">try again</a>.</div>';

exit();

}

elseif (strlen($results\_maxgrade) > 3)

{

print '<div class="alert alert-danger"><strong>Your total score is more than the three-digit number!</strong> You should <a href="evaluate.php" class="alert-link">try again</a>.</div>';

exit();

}

elseif (strlen($results\_comment) > 250)

{

print '<div class="alert alert-danger"><strong>Your comment is longer than 250 characters!</strong> You should <a href="evaluate.php" class="alert-link">try again</a>.</div>';

exit();

}

*// This SQL statement looks for the work of a specific student that the teacher intends to grade*

$query = mysqli\_query($conn, “SELECT \* FROM work WHERE work\_check = 0 AND ‘$userID’ = userID”);

$ar = mysqli\_fetch\_array($query);

*// If the teacher assigned the wrong workID to the student, then the program will give an error. This condition is necessary to prevent the situation when the teacher evaluated a non-existent document or a document that belongs to another student*

if ($workID == $ar[“workID”])

{

$workID = $\_POST[“workID”];

*// Insertion of the data entered by the user into one of the database tables*

$add = “INSERT INTO results (userID, workID, results\_grade, results\_maxgrade, results\_comment) VALUES (‘$userID’, ‘$workID’, ‘$results\_grade’, ‘$results\_maxgrade’, ‘$results\_comment’)”;

*// Change the work\_check value to 1 to indicate that the student’s work has been graded and does not need to be displayed for grading*

$add1 = mysqli\_query($conn,”UPDATE work SET work\_check = 1 WHERE workID = ‘$workID’”);

// This condition checks if the user data was inserted successfully

if (mysqli\_query($conn, $add))

{

print ‘<div class=”alert alert-success”><strong>This work has been evaluated uccessfully!</strong> Go back to <a href=”main2.php” class=”alert-link”>home page</a>.</div>’;

}

else

{

print ‘<div class=”alert alert-danger”><strong>Error!</strong> You should <a href=”main1.php” class=”alert-link”>try again</a>.</div>’;

exit();

}

}

else

{

print ‘<div class=”alert alert-danger”><strong>Error!</strong> You have written an wrong full name or workID! You should <a href=”main1.php” class=”alert-link”>try again</a>.</div>’;

exit();

}

}

?>

**The “Output work” form**

Admin

The screenshot below shows the output of student work if the user is a teacher. The teacher has access to all students’ files and has an opportunity to evaluate them.

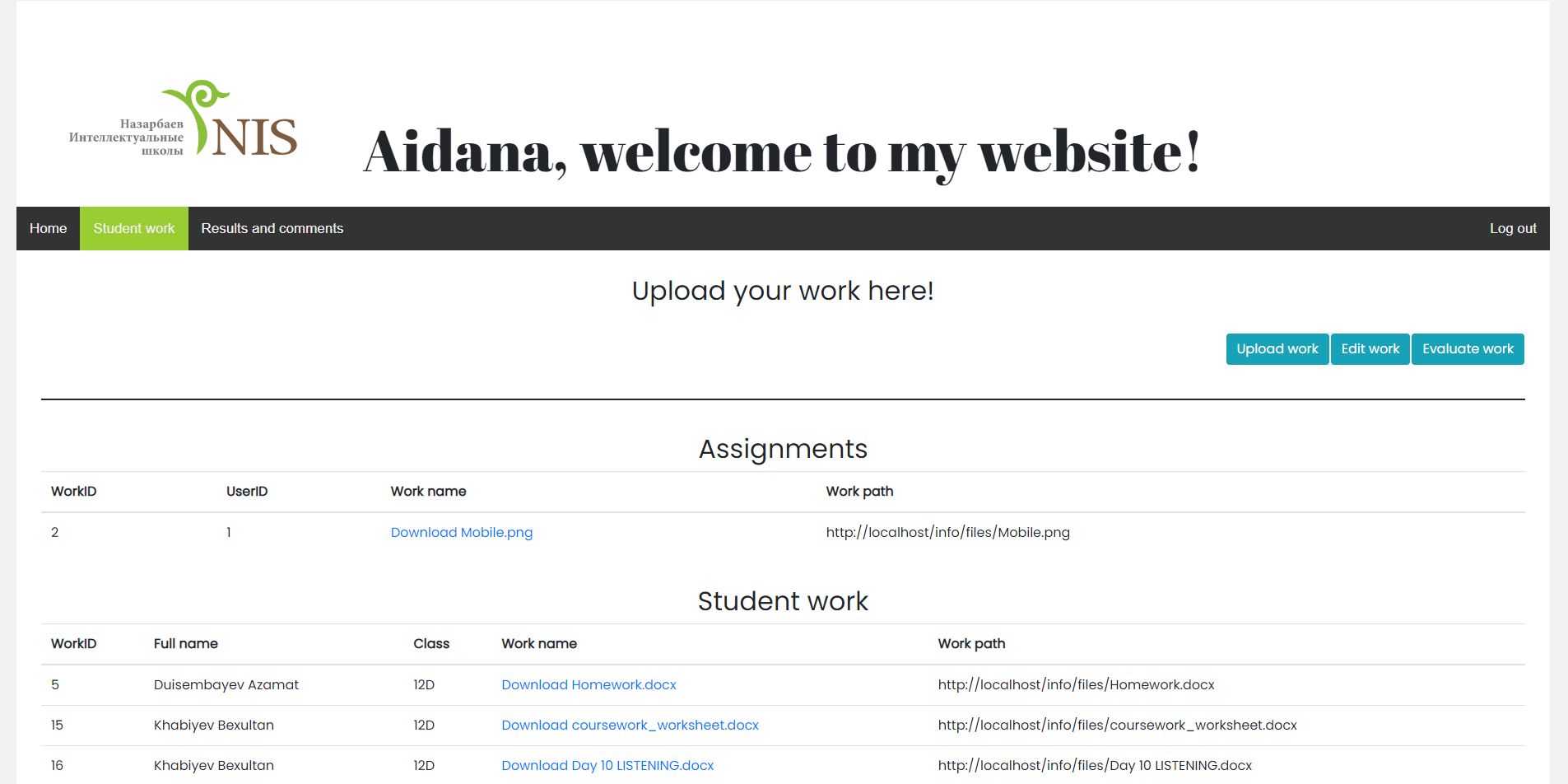


Figure 35. Work Output form (admin)

User

The screenshot below shows the output of student work if the user is a student. The student has access only to his works to prevent discrimination against other students based on academic performance.

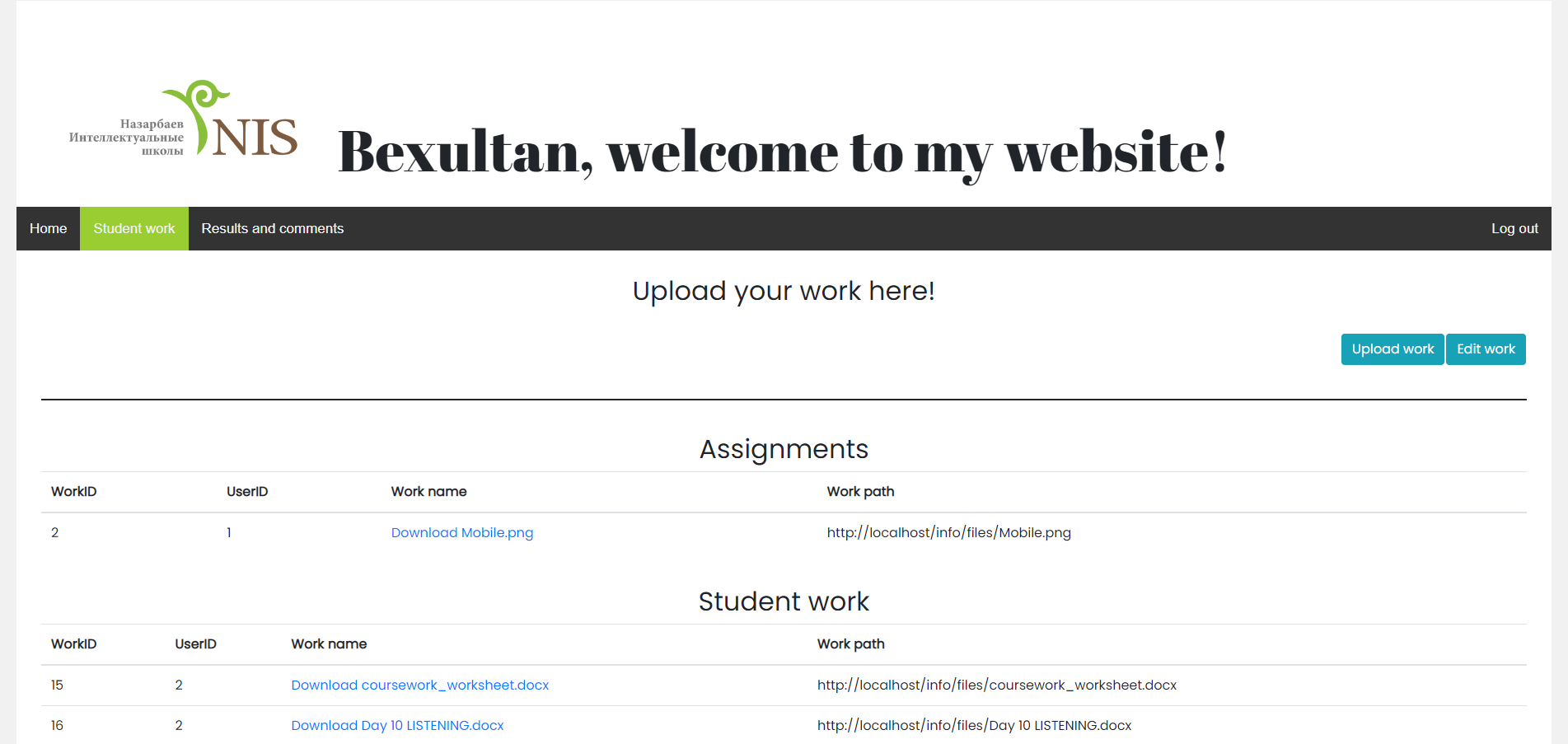
****

Figure 36. Work Output form (user)

File “handlermain1.php”

<?php

*// Database connection*

include(‘connect.php’);

*// We set the path to the file as a variable to shorten the length of the program code*

$currentfile = ‘http://localhost/info/files/’;

*// Setting variables for user data stored in the session*

**$user\_status = $\_SESSION[‘data’][‘user\_status’];**

$userID = $\_SESSION[‘data’][‘userID’];

*// These conditions check the status of the current user. This is necessary so that the student cannot see the work of other students. Only a user with the status “teacher” has access to all files*

if ($user\_status == ‘teacher’)

{

*// This request contains the data of the work that the teacher uploaded*

$search = mysqli\_query($conn, “SELECT userID FROM users WHERE user\_status = ‘teacher’”);

$userID1 = mysqli\_fetch\_array($search);

$myrow1 = mysqli\_query($conn, “SELECT \* FROM work WHERE work\_check = 0 AND userID =”.$userID1[“userID”].””);

*// These loops form a table that contains the files that the teacher has uploaded. This is done so that the student can quickly find the assignment*

echo ‘<h2 align=”center”>Assignments</h2>’;

echo ‘<table class=”table table-hover”>’;

echo ‘<thead><tr><th>WorkID</th><th>UserID</th><th>Work name</th><th>Work path</th></tr></thead>’;

echo ‘<tbody>’;

*// These loops display all records that contain the data specified in mysqli request. The loops run as long as there is data to be output in the variable. They allow programmist to shorten and automate the program code, since there is no need to regularly enter each row of the table*

While ($row = mysqli\_fetch\_array($myrow1))

{

echo ‘<tr>’;

echo ‘<td>’. $row[‘workID’] .’</td>’;

echo ‘<td>’. $row[‘userID’] .’</td>’;

echo ‘<td><a href=”’. $currentfile.$row[‘work\_name’] .’”>Download ‘.$row[‘work\_name’].’</a></td>’;

echo ‘<td>’. $currentfile.$row[‘work\_name’] .’</td>’;

echo ‘</tr>’;

}

echo ‘</tbody>’;

echo ‘</table>’;

echo ‘<br>’;

*// This request contains data intended for teacher. The request includes only works without grades*

$myrow = mysqli\_query($conn, “SELECT \*, users.user\_name, users.user\_surname, users.user\_class FROM work LEFT JOIN users ON work.userID = users.userID WHERE work\_check = 0 AND work.userID !=”.$userID1[“userID”].””);

echo ‘<h2 align=”center”>Student work</h2>’;

echo ‘<table class=”table table-hover”>’;

echo ‘<thead><tr><th>WorkID</th><th>Full name</th><th>Class</th><th>Work name</th><th>Work path</th></tr></thead>’;

echo ‘<tbody>’;

while ($row = mysqli\_fetch\_array($myrow))

{

echo ‘<tr>’;

echo ‘<td>’. $row[‘workID’] .’</td>’;

echo ‘<td>’. $row[‘user\_surname’], ‘ ‘, $row[‘user\_name’] .’</td>’;

echo ‘<td>’. $row[‘user\_class’] .’</td>’;

echo ‘<td><a href=”’. $currentfile.$row[‘work\_name’] .’”>Download ‘.$row[‘work\_name’].’</a></td>’;

echo ‘<td>’. $currentfile.$row[‘work\_name’] .’</td>’;

echo ‘</tr>’;

}

echo ‘</tbody>’;

echo ‘</table>’;

}

elseif ($user\_status == ‘student’)

{

$search = mysqli\_query($conn, “SELECT userID FROM users WHERE user\_status = ‘teacher’”);

$userID1 = mysqli\_fetch\_array($search);

$myrow = mysqli\_query($conn, “SELECT \* FROM work WHERE work\_check = 0 AND userID =”.$userID1[“userID”].””);

echo ‘<h2 align=”center”>Assignments</h2>’;

echo ‘<table class=”table table-hover”>’;

echo ‘<thead><tr><th>WorkID</th><th>UserID</th><th>Work name</th><th>Work path</th></tr></thead>’;

echo ‘<tbody>’;

while ($row = mysqli\_fetch\_array($myrow))

{

echo ‘<tr>’;

echo ‘<td>’. $row[‘workID’] .’</td>’;

echo ‘<td>’. $row[‘userID’] .’</td>’;

echo ‘<td><a href=”’. $currentfile.$row[‘work\_name’] .’”>Download ‘.$row[‘work\_name’].’</a></td>’;

echo ‘<td>’. $currentfile.$row[‘work\_name’] .’</td>’;

echo ‘</tr>’;

}

echo ‘</tbody>’;

echo ‘</table>’;

echo ‘<br>’;

*// This request contains data intended only for a specific user. The request includes only works without grades*

$myrow1 = mysqli\_query($conn, “SELECT \* FROM work WHERE work\_check = 0 AND userID =’$userID’”);

echo ‘<h2 align=”center”>Student work</h2>’;

echo ‘<table class=”table table-hover”>’;

echo ‘<thead><tr><th>WorkID</th><th>UserID</th><th>Work name</th><th>Work path</th></tr></thead>’;

echo ‘<tbody>’;

while ($row = mysqli\_fetch\_array($myrow1))

{

echo ‘<tr>’;

echo ‘<td>’. $row[‘workID’] .’</td>’;

echo ‘<td>’. $row[‘userID’] .’</td>’;

echo ‘<td><a href=”’. $currentfile.$row[‘work\_name’] .’”>Download ‘.$row[‘work\_name’].’</a></td>’;

echo ‘<td>’. $currentfile.$row[‘work\_name’] .’</td>’;

echo ‘</tr>’;

}

echo ‘</tbody>’;

echo ‘</table>’;

}

?>

**The “Results and comments output” form**

Admin

The screenshot below shows the output of student results if the user is a teacher. The teacher has access to all students’ results.

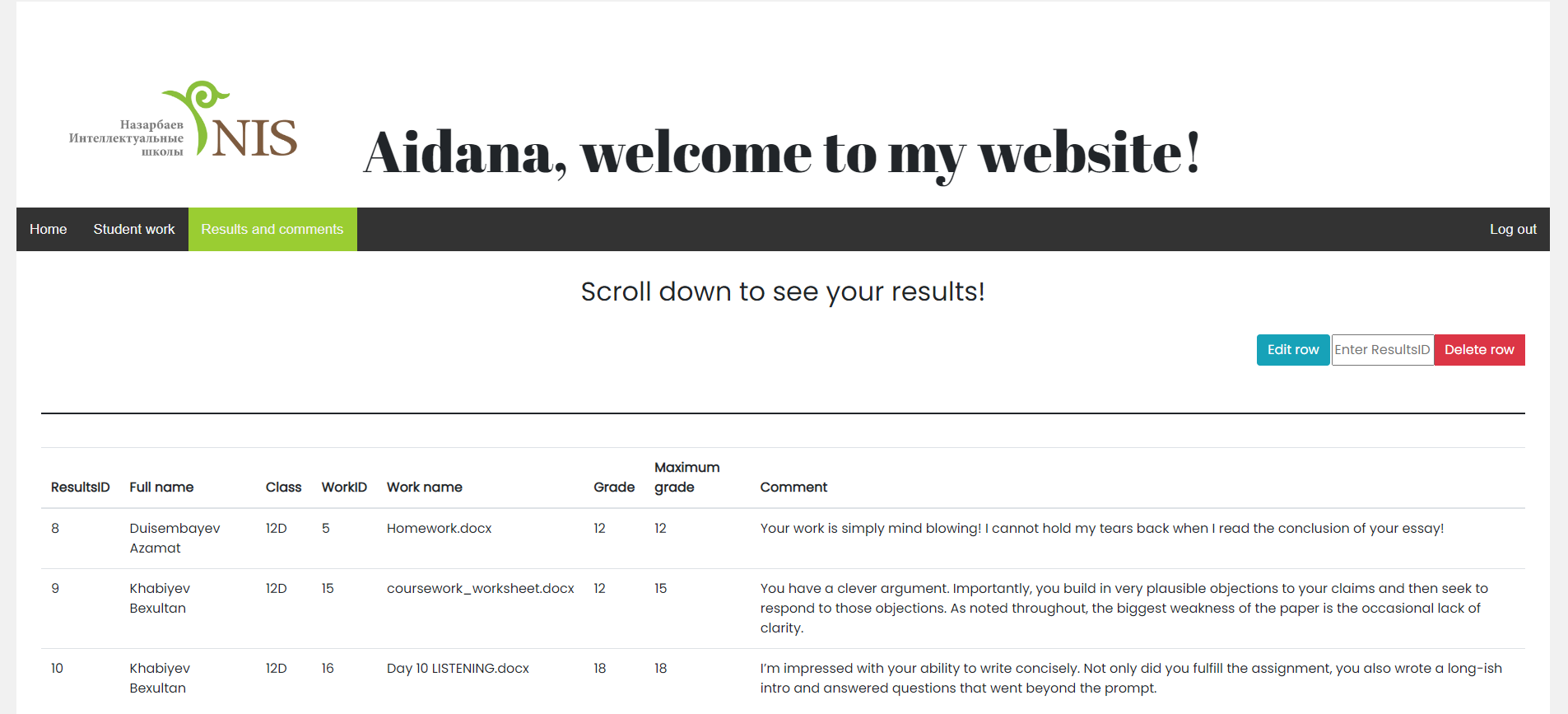


Figure 37. Results and Comment Output form (admin)

User

The presented screenshot shows the output of student results if the user is a student. The student has access only to his own results in order to prevent discrimination against other students according to academic activities.

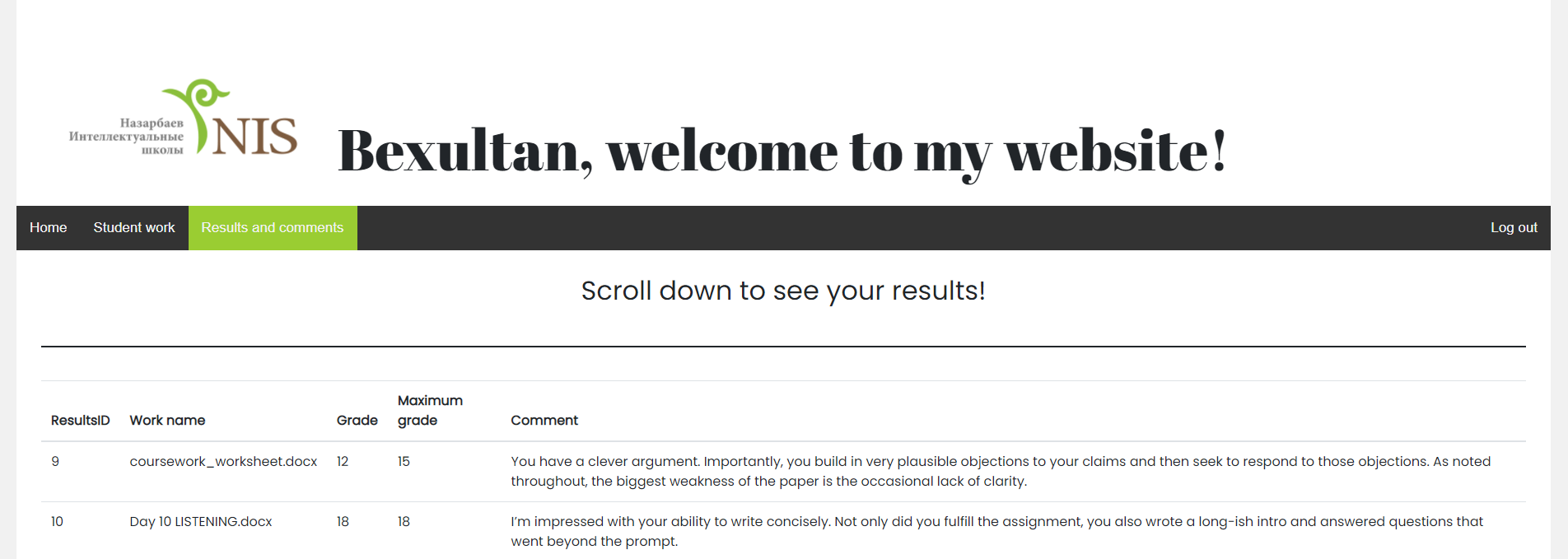


Figure 38. Results and Comment Output form (user)

File “handlermain2.php”

<?php

*// Database connection*

include(‘connect.php’);

*// Setting variables for user data stored in the session*

$user\_status = $\_SESSION[‘data’][‘user\_status’];

$userID = $\_SESSION[‘data’][‘userID’];

*// These conditions check the status of the current user. This is necessary so that the student cannot see the work of other students. Only a user with the status “teacher” has access to all files*

if ($user\_status == ‘teacher’)

{

*// This request contains data intended for teacher*

**$myrow = mysqli\_query($conn, “SELECT \*, users.user\_surname, users.user\_class, users.user\_name, work.work\_name FROM results LEFT JOIN users ON results.userID = users.userID LEFT JOIN work ON results.workID = work.workID ORDER BY results.resultsID”);**

echo ‘<table class=”table table-hover”>’;

echo ‘<thead><tr><th>ResultsID</th><th>Full name</th><th>Class</th><th>Work name</th><th>Grade</th><th>Maximum grade</th><th>Comment</th></tr></thead>’;

echo ‘<tbody>’;

*// These loops displays all records that contain the data specified in mysqli request. The loops run as long as there is data to be output in the variable. They allow programmist to shorten and automate the program code, thus there is no need to regularly enter each row of the table*

while($row = mysqli\_fetch\_array($myrow))

{

echo ‘<tr>’;

echo ‘<td>’. $row[‘resultsID’] .’</td>’;

echo ‘<td>’. $row[‘user\_surname’], ‘ ‘, $row[‘user\_name’] .’</td>’;

echo ‘<td>’. $row[‘user\_class’] .’</td>’;

echo ‘<td>’. $row[‘work\_name’] .’</td>’;

echo ‘<td>’. $row[‘results\_grade’] .’</td>’;

echo ‘<td>’. $row[‘results\_maxgrade’] .’</td>’;

echo ‘<td>’. $row[‘results\_comment’] .’</td>’;

echo ‘</tr>’;

}

echo ‘</tbody>’;

echo ‘</table>’;

}

elseif ($user\_status == ‘student’)

{

*// This request contains data intended only for a specific user*

**$myrow1 = mysqli\_query($conn, “SELECT \*, work.work\_name FROM results LEFT JOIN work ON results.workID = work.workID WHERE results.userID =’$userID’”);**

echo ‘<table class=”table table-hover”>’;

echo ‘<thead><tr><th>ResultsID</th><th>Work name</th><th>Grade</th><th>Maximum grade</th><th>Comment</th></tr></thead>’;

echo ‘<tbody>’;

while($row = mysqli\_fetch\_array($myrow1))

{

echo ‘<tr>’;

echo ‘<td>’. $row[‘resultsID’] .’</td>’;

echo ‘<td>’. $row[‘work\_name’] .’</td>’;

echo ‘<td>’. $row[‘results\_grade’] .’</td>’;

echo ‘<td>’. $row[‘results\_maxgrade’] .’</td>’;

echo ‘<td>’. $row[‘results\_comment’] .’</td>’;

echo ‘</tr>’;

}

echo ‘</tbody>’;

echo ‘</table>’;

}

?>

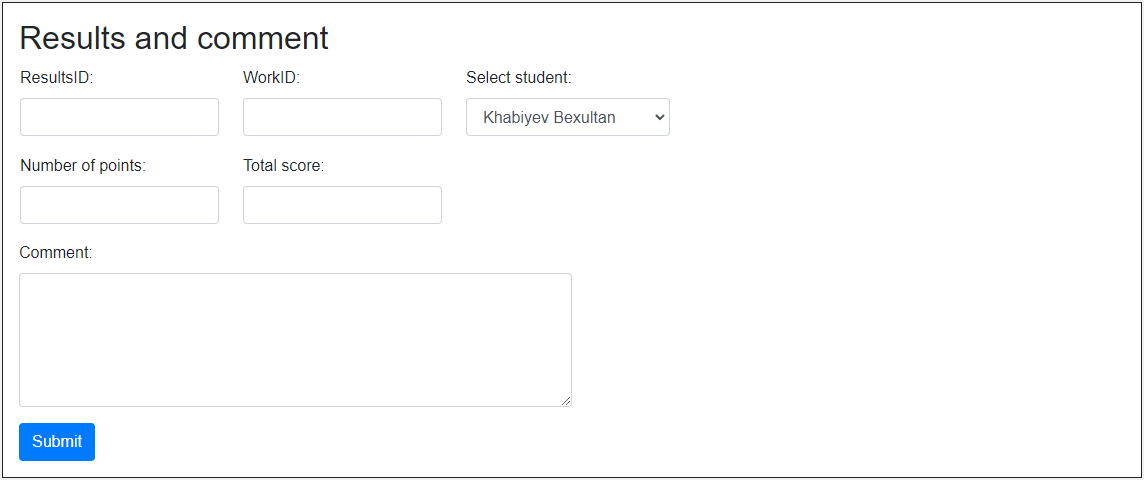
**The “Edit results” form**

Figure 39. Results Editting form

File “edit1.php”

<form action="edit1handler.php" method="POST">

<table>

<tr>

<td>

<div class="form-group" style="width: 90%;">

<label for="workID">ResultsID:</label>

<input type="number" class="form-control" name="resultsID" required>

</div>

</td>

<td>

<div class="form-group" style="width: 90%;">

<label for="workID">WorkID:</label>

<input type="number" class="form-control" name="workID" required>

</div>

</td>

<td>

<div class="form-group">

<label for="fullname">Select student:</label>

<select name="fullname" class="form-control">

<?php

include('connect.php');

$myrow = mysqli\_query($conn, "SELECT \* FROM users WHERE user\_status ='student'");

while ($row = mysqli\_fetch\_array($myrow))

{

echo "<option>". $row['user\_surname'], ' ', $row['user\_name'] ."</option>";

}

?>

</select>

</div>

</td>

</tr>

</table>

<table>

<tr>

<td>

<div class="form-group" style="width: 90%;">

<label for="grade">Number of points:</label>

<input type="number" class="form-control" name="grade" required>

</div>

</td>

<td>

<div class="form-group" style="width: 90%;">

<label for="maxgrade">Total score:</label>

<input type="number" class="form-control" name="maxgrade" required>

</div>

</td>

</tr>

</table>

<div class="form-group" style="width: 50%;">

<label for="comment">Comment:</label>

<textarea class="form-control" rows="5" name="comment" required></textarea>

</div>

<div class="mt-3">

<button type="submit" class="btn btn-primary">Submit</button>

</div>

</form>

File “edit1handler.php”

<?php

*// Database connection*

include ("connect.php");

*// Transfer data from the form and define variables for it*

$user\_fullname = $\_POST["fullname"];

*// This SQL statement looks for users with a full name similar to the full name obtained from the form*

$searchfullname = mysqli\_query($conn, "SELECT userID FROM users WHERE '$user\_fullname' = CONCAT(user\_surname, ' ', user\_name)");

$myrow = mysqli\_fetch\_array($searchfullname);

*// If there is a user ID, where his full name matches the full name entered by the user, then the program will continue executing the algorithm*

if (!empty($myrow['userID']))

{

*// Transfer data from the form and define variables for it*

$resultsID = $\_POST["resultsID"];

$userID = $myrow["userID"];

$workID = $\_POST["workID"];

$results\_grade = $\_POST["grade"];

$results\_maxgrade = $\_POST["maxgrade"];

$results\_comment = $\_POST["comment"];

*// These conditions check the length of the entered data. If the number of characters doesn't fill in the allowed range, then the program will generate an error*

if (strlen($results\_grade) > 3)

{

print '<div class="alert alert-danger"><strong>Your number of points is more than the three-digit number!</strong> You should <a href="evaluate.php" class="alert-link">try again</a>.</div>';

exit();

}

elseif (strlen($results\_maxgrade) > 3)

{

print '<div class="alert alert-danger"><strong>Your total score is more than the three-digit number!</strong> You should <a href="evaluate.php" class="alert-link">try again</a>.</div>';

exit();

}

elseif (strlen($results\_comment) > 250)

{

print '<div class="alert alert-danger"><strong>Your comment is longer than 250 characters!</strong> You should <a href="evaluate.php" class="alert-link">try again</a>.</div>';

exit();

}

*// This SQL statement looks for the work of a specific student that the teacher intends to grade*

$query = mysqli\_query($conn, "SELECT \* FROM work WHERE work\_check = 1 AND '$userID' = userID");

$ar = mysqli\_fetch\_array($query);

*// If the teacher assigned the wrong workID to the student, then the program will give an error. This condition is necessary to prevent the situation when the teacher evaluated a non-existent document or a document that belongs to another student*

if ($workID == $ar["workID"])

{

$searchworkID = mysqli\_query($conn, "SELECT \* FROM results WHERE resultsID = '$resultsID'");

$ar1 = mysqli\_fetch\_array($searchworkID);

*// This condition checks if the teacher wants to change the assessment results for a particular student or assign the assessment results to another student. This is necessary in case when the teacher initially wrote the workID and the name of the wrong student when evaluating the work*

if ($workID == $ar1["workID"])

{

*// Replacing the data that is in the table "results" with the data entered by the teacher*

$add = "UPDATE results SET userID = '$userID', workID = '$workID', results\_grade = '$results\_grade', results\_maxgrade = '$results\_maxgrade', results\_comment = '$results\_comment' WHERE resultsID = '$resultsID'";

}

else

{

*// This line marks the work, which is replaced by another, as unassessed, so that the teacher can check it in the future*

$add0 = mysqli\_query($conn,"UPDATE work SET work\_check = 0 WHERE workID = ".$ar1["workID"]."");

*// Replacing the data that is in the table "results" with the data entered by the teacher*

$add = "UPDATE results SET userID = '$userID', workID = '$workID', results\_grade = '$results\_grade', results\_maxgrade = '$results\_maxgrade', results\_comment = '$results\_comment' WHERE resultsID = '$resultsID'";

*// This line marks new work as reviewed*

$add1 = mysqli\_query($conn,"UPDATE work SET work\_check = 1 WHERE workID = '$workID'");

}

*// This condition checks if the user data was inserted successfully*

if (mysqli\_query($conn, $add))

{

print '<div class="alert alert-success"><strong>This row was successfully edited!</strong> Go back to <a href="main2.php" class="alert-link">home page</a>.</div>';

}

else

{

print '<div class="alert alert-danger"><strong>Error!</strong> You should <a href="main1.php" class="alert-link">try again</a>.</div>';

exit();

}

}

else

{

print '<div class="alert alert-danger"><strong>Error!</strong> You have written a wrong full name, workID or resultsID! You should <a href="main2.php" class="alert-link">try again</a>.</div>';

exit();

}

}

?>

**The “Delete results” form**

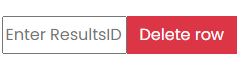
****

Figure 40. Results Deleting form

File “main2.php”

<form action="deletehandler.php" method="POST">

<div class="input-group mb-3">

<input type="text" style="width: 125px" placeholder="Enter ResultsID" name="resultsID">

<div class="input-group-prepend">

<button class="btn btn-danger" type="submit">Delete row</button>

</div>

</div>

</form>

File “deletehandler.php”

<?php

*// Database connection*

include ("connect.php");

*// Transfer data from the form and define variables for it*

$resultsID = $\_POST["resultsID"];

$add = mysqli\_query($conn,"SELECT \* FROM results WHERE resultsID = '$resultsID'");

$ar = mysqli\_fetch\_array($add);

*// Delete selected row from two linked tables: results and work*

$wreckingcrew = mysqli\_query($conn,"DELETE FROM results WHERE resultsID = '$resultsID'");

$wreckingcrew1 = mysqli\_query($conn,"DELETE FROM work WHERE workID = ".$ar["workID"]."");

header('Location: main2.php');

?>

## Testing

In order to check the new system for existing errors and make sure that the website is working correctly, testing has been carried out. Testing the new system involves checking the following website functions:

Registration

Authorization

Uploading files

Editing work

Evaluation of uploaded files

Output of student work files

Output of test results and teacher comments

Editing results

Deleting results

Admin mode

User mode

Test plan

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **#** | **Purpose** | **Description** | **Test data** | | | | **Expected result** | **Actual result** |
| 1.1 | Check the collection of input data that needs to be entered in order to register a new user | Don’t fill in one of the fields or all the fields required to register a new user, excluding the class field | Name:“”  Surname:“”  Login:“”  Password:“”  Repeat password:“”  Class:“”  Status:“” | | | | Show the message “Fill in this field” | True |
| Evidence 1.1 | | | | | | | | |
| 1.2 | Check the collection of input data that needs to be entered in order to register a new user | Fill in all fields for registration of a new user, excluding status | Name:“Chuck”  Surname:“Shuldiner”  Login:“ilovemetal123”  Password:“ilovemetal”  Repeat password:“ilovemetal”  Class:“12D”  Status:“” | | | | Show the message “You must enter your status!” | True |
| Evidence 1.2 | | | | | | | | |
| 1.3 | Check the collection of input data that needs to be entered in order to register a new user | Don’t fill in the class field if the user has selected student status | Name:“Chuck”  Surname:“Shuldiner”  Login:“metalrules123”  Password:“ilovemetal”  Repeat password:“ilovemetal”  Class:“”  Status:“Student” | | | | Show the message “You must enter your class name!” | True |
| Evidence 1.3 | | | | | | | | |
| 1.4 | Check the collection of input data that needs to be entered in order to register a new user | Don’t fill in the class field if the user has selected teacher status | Name:“Chuck”  Surname:“Shuldiner”  Login:“ilovemetal123”  Password:“ilovemetal”  Repeat password:“ilovemetal”  Class:“”  Status:“Teacher” | | | | Show the message “You have successfully registered!” | True |
| Evidence 1.4 | | | | | | | | |
| 1.5 | Check the collection of input data that needs to be entered in order to register a new user | Fill in all the fields with valid values | Name:“Chuck”  Surname:“Shuldiner”  Login:“ilovemetal123”  Password:“ilovemetal”  Repeat password:“ilovemetal”  Class:“12D”  Status:“Student” | | | | Show the message “You have successfully registered!” | True |
| Evidence 1.5 | | | | | | | | |
| 1.6 | Check the collection of input data that needs to be entered in order to register a new user | Fill the login field with extreme values | Name:“Chuck”  Surname:“Shuldiner”  Login:“aaaaaaaa”  Password:“ilovemetal”  Repeat password:“ilovemetal”  Class:“12D”  Status:“Student” | | | | Show the message “You have successfully registered!” | True |
| Evidence 1.6 | | | | | | | | |
| 1.7 | Check the collection of input data that needs to be entered in order to register a new user | Fill the login field with erroneous values | Name:“Chuck”  Surname:“Shuldiner”  Login:“123”  Password:“ilovemetal”  Repeat password:“ilovemetal”  Class:“12D”  Status:“Student” | | | | Show the message “Your login is longer than 30 characters or shorter than 8 characters!” | True |
| Evidence 1.7 | | | | | | | | |
| 1.8 | Check the verification process of the existing login entered by the user | Fill the login field with an existing login in the database | Name:“Chuck”  Surname:“Shuldiner”  Login:“bekakrut”  Password:“ilovemetal”  Repeat password:“ilovemetal”  Class:“12D”  Status:“Student” | | | | Show the message “Your login already exists!” | True |
| Evidence 1.8 | | | | | | | | |
| 1.9 | Check the verification process of a repeated password that was written incorrectly | Repeat the entered password incorrectly | Name:“Chuck”  Surname:“Shuldiner”  Login:“ ilovemetal1234”  Password:“ilovemetal”  Repeat password:“ilovemetal228”  Class:“12D”  Status:“Student” | | | | Show the message “Passwords do not match!” | True |
| Evidence 1.9 | | | | | | | | |
| 2.1 | Check the collection of input data that needs to be entered during the log in process | Don’t fill in the login field and password field or leave one of them empty | Login:“”  Password:“” | | | Show the message “Fill in this field!” | | True |
| Evidence 2.1 | | | | | | | | |
| 2.2 | Check the collection of input data that needs to be entered during the log in process | Enter the username and password that don’t exist in the database | Login:“aaa”  Password:“ooo” | | | Show the message “Your login or password was entered incorrectly!” | | True |
| Evidence 2.2 | | | | | | | | |
| 2.3 | Check the collection of input data that needs to be entered during the log in process | Enter the username and password that exist in the database | Login:“ilovemetal123”  Password:“ilovemetal” | | | Show the message “You have successfully entered!” | | True |
| Evidence 2.3 | | | | | | | | |
| 3.1 | Check the process of loading a file into the system database | Do not upload file to input field | | Custom file:“” | Show the message “You have not uploaded your file!” | | | True |
| Evidence 5.1 | | | | | | | | |
| 3.2 | Check the process of loading a file into the system database | Upload file into input field | | Custom file:“Homework.docx” | Show the message “Your file was successfully uploaded!” | | | True |
| Evidence 3.2 | | | | | | | | |
| 4.1 | Check the process of editing work files that have been uploaded to the system | Do not upload file to input field | | WorkID:“5”  Custom file:“” | Show the message “You have not uploaded your file!” | | | True |
| Evidence 4.1 | | | | | | | | |
| 4.2 | Check the process of editing work files that have been uploaded to the system | Write WorkID of a work file that belongs to another student or does not exist | | WorkID:“37”  Custom file:“Homework.docx” | Show the message “You are trying to edit work of another student or that does not exist!” | | | True |
| Evidence 4.3 | | | | | | | | |
| 4.3 | Check the process of editing work files that have been uploaded to the system | Upload another file to the system in place of the existing one | | WorkID:“5”  Custom file:“Homework.docx” | Show the message “Your file was successfully uploaded!” | | | True |
| Evidence 4.3 | | | | | | | | |
| 5.1 | Check the process of setting the results by the teacher for the work of the student | Don’t fill in one of the fields or all the fields required to evaluate the student’s work | | WorkID:“”  Select student:“Khabiyev Bexultan”  Number of points: “”  Total score:“”  Comment:“” | Show the message "Fill in this field" | | | True |
| Evidence 5.1 | | | | | | | | |
| 5.2 | Check the process of setting the results by the teacher for the work of the student | Fill in all fields with valid values | | WorkID:“12”  Select student:“Chuck Shuldiner”  Number of points: “8”  Total score:“10”  Comment:“Overall, the research you have conducted on the reproduction of germs was excellent. However, you have neglected several significant details.” | Show the message “The work has been evaluated successfully” | | | True |
| Evidence 5.2 | | | | | | | | |
| 5.3 | Check the process of setting the results by the teacher for the work of the student | Fill the total score field with extreme values | | WorkID:“8”  Select student:“Khabiyev Bexultan”  Number of points: “135”  Total score:“140”  Comment:“I was amazed by your profound knowledge of the subject! Keep up the great work!” | Show the message "Fill in this field" | | | True |
| Evidence 5.3 | | | | | | | | |
| 5.4 | Check the process of setting the results by the teacher for the work of the student | Fill the total score field with erroneous values | | WorkID:“8”  Select student:“Khabiyev Bexultan”  Number of points: “100”  Total score:“1500”  Comment:“I was amazed by your profound knowledge of the subject! Keep up the great work!” | Show the message “Your total score is more than the three-digit number!” | | | True |
| Evidence 5.4 | | | | | | | | |
| 5.5 | Check the process of setting the results by the teacher for the work of the student | Assign a workID that does not belong to the selected student or select an incorrect student from the drop-down list | | WorkID:“23”  Select student:“Khabiyev Bexultan”  Number of points: “12”  Total score:“12”  Comment:“I really appreciate the amount of work you have done!” | Show the message “You have written a wrong full name or workID!” | | | True |
| Evidence 5.5 | | | | | | | | |
| 6.1 | Check the editing process for results that have already been posted | Don’t fill in one of the fields or all the fields required to edit results | | ResultsID:“2”  WorkID:“2”  Select student:“Khabiyev Bexultan”  Number of points:“2”  Total score:“2”  Comment:“” | Show the message “You missed this field.” | | | True |
| Evidence 6.1 | | | | | | | | |
| 6.2 | Check the editing process for results that have already been posted | Write ResultsID that does not exist in the table | | ResultsID:“25”  WorkID:“6”  Select student:“Uteuliyev Assanali”  Number of points:“25”  Total score:“25”  Comment:“I was quite impressed by your work!” | Show the message “You have written a wrong full name, workID or resultsID!” | | | True |
| Evidence 6.2 | | | | | | | | |
| 6.3 | Check the editing process for results that have already been posted | Change the maximum number of points in the student's results and rewrite a comment | | ResultsID:“1”  WorkID:“6”  Select student:“Uteuliyev Assanali”  Number of points:“23”  Total score:“30”  Comment:“I am sure you could do better.” | Show the message “This row was successfully edited!” | | | True |
| Evidence 6.3 | | | | | | | | |
| 7.1 | Check the process of deleting rows in the "Results" table | Write the resultsID of the row to be deleted | | ResultsID:“7” | The specified table row will be deleted | | | True |
| Evidence | | | | | | | | |
| 8.1 | Check the system's reaction to an attempt to view student work by an unauthorized user | Do not log in and try to see the work of students | | - | Show the message “You must log in first” | | | True |
| Evidence 8.1 | | | | | | | | |
| 8.2 | Check the system's reaction to an attempt to view student results by an unauthorized user | Do not log in and do try to see the results of students | | - | Show the message “You must log in first” | | | True |
| Evidence 8.2 | | | | | | | | |
| 9.1 | Check the output of students' work if the user is a teacher | Log in as a teacher and view student work | | - | Display a table with the work of all students | | | True |
| Evidence 9.1 | | | | | | | | |
| 9.2 | Check the output of the results of all students for the completed work if the user is a teacher | Log in as a teacher and view student results | | - | Display a table with the results of all students | | | True |
| Evidence 9.2 | | | | | | | | |
| 10.1 | Check the output of the student work if the user is a student | Log in as a student and view student work | | - | Display a table with the work of one student who entered the website | | | True |
| Evidence 10.1 | | | | | | | | |
| 10.2 | Check the output of student results for completed work if the user is a student | Log in as a user and view student results | | - | Display a table with the results of one student who entered the website | | | True |
| Evidence 10.2 | | | | | | | | |
| 11.1 | Check if the navigation bar changes after user authorization | Log in as an authorized user | | - | The log out option will appear, while the registration and authorization options will be removed | | | True |
| Evidence 11.1 | | | | | | | | |

## Installation

**Installation plan**

To use the new system’s capabilities most effectively, it is necessary to explain how the system works to my client. I discussed the date with the client for the installation of the system and thus, the following installation plan was developed:

* + - 1. Show the client a web page to convince the client that it is working correctly and meets the client's expectations
      2. Demonstrate the main functions of the website and explain how the student assessment process should proceed
      3. Allow the client to try out all the features of the website on her own to make sure that the client understands how to work with the new system
      4. Keep in touch with the client to help him in case the client has questions regarding the use of the new system
      5. Implement a new system for permanent use by the client

**Training plan**

Study date:

27 November

Reason: This date falls on Saturday, which is a non-working day. My client is a teacher at school who will not have enough free time on weekdays, as he will be busy at his workplace. For this reason, Saturday is the right day to explain how the new system works and show all its possibilities. In addition, if my client has any questions about the website, we can discuss them the next day, Sunday, because it is also a non-working day.

Method of study: It is planned that the study will take place within one day, if necessary; it will be possible to continue training the next day. The training will take two hours and will be held online on the Teams platform since my client lives in another city and has an already registered account in this application.  First of all, I will show the client how to work with the new system and what functions the website has. I will then give the client an opportunity to try out all the features of the new system to make sure she understands how to work with the new system.

**System installation plan**

Method: direct changeover

Reason: The new system, like the previous one, is aimed at conducting the summative assessment, so the changeover is the most logical method of implementation. Direct changeover allows the system to be installed quickly compared to other methods, which is very important for the immediate continuation of the academic assessment of students in the school environment. Parallel installation cannot be an implementation method, since in this case, students have to complete the summative assessment tasks twice: in paper and electronic format, in order to support both systems. In addition, my client is already familiar with various applications as she has experience in distance learning. This will allow her to shift to the new system easily.

Installation date:

25 December

Reason: This day is included in the school holidays. For this reason, my client and I will be free at this time, which will make it possible to install a new system. Moreover, the time between the study date and the date of installation is sufficient for the client to explore the website independently.

**The evidence of user test**

In order to make sure that the client agrees with the installation plan and training plan, we set an online meeting and discussed all the details of these two plans such as the date and the amount of time required to implement them. During the online meeting, the client had the opportunity to test the capabilities of the new system and evaluate the appearance of the website. After that, the client filled out the questionnaire that I submitted to summarize the client's feedback. The questionnaire allowed me to understand what the client liked and what did not like about the work of the website in a short period.  
I needed to receive an email from a client that confirms that she can meet with me online to test the system according to the training plan and carry out its installation.



Figure 41. Letter confirming the client's agreement with the terms of the test plan

**The results of the questionnaire**

After the client had reviewed the website and tested the capabilities of the new system, he gave feedback by filling out the survey form.

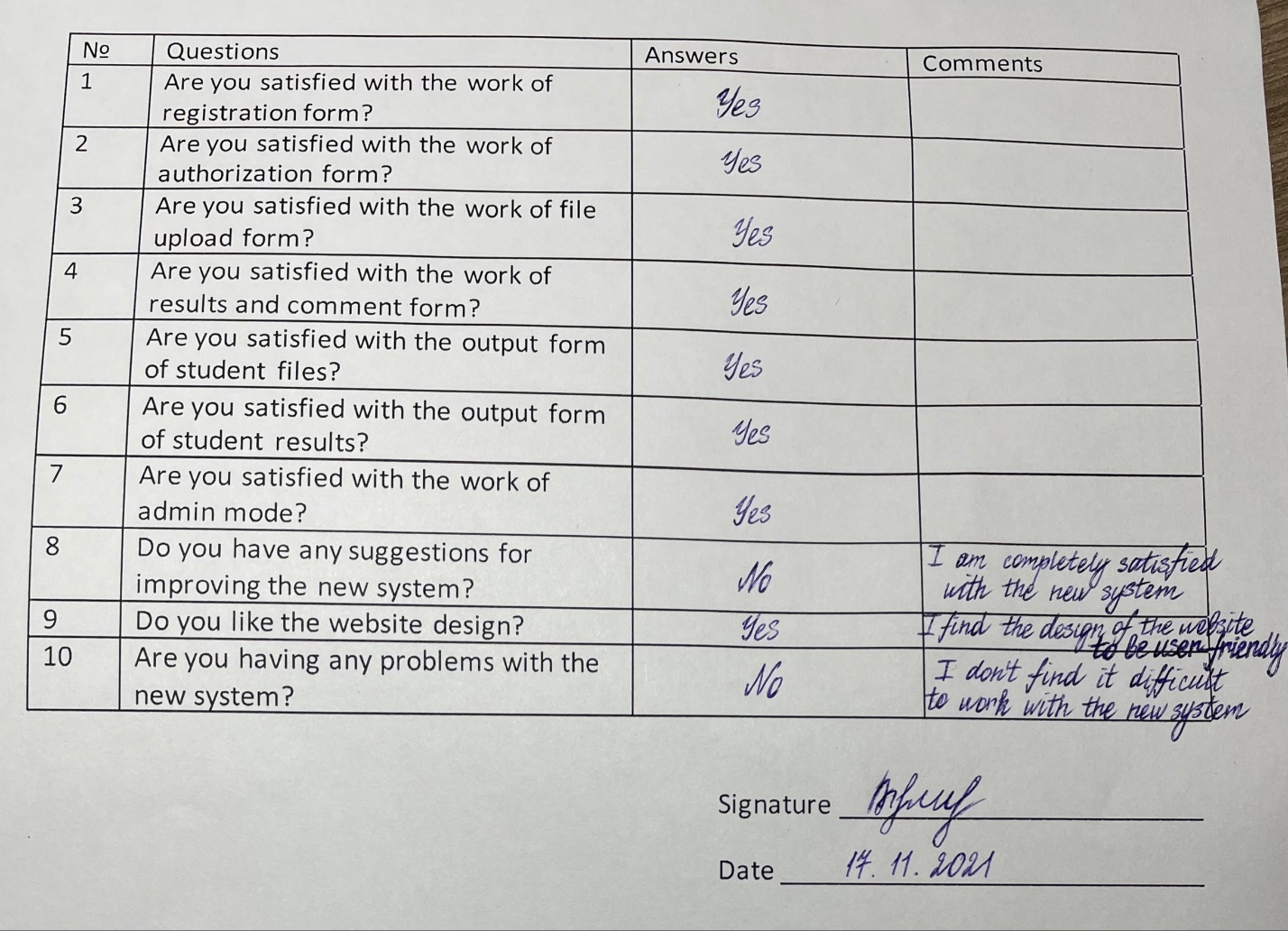
****

Figure 42. Client survey responses about the new system

# Documentation

## User guide

Table of content:

Registration

Authorization

Uploading a work file to a website

Student works output

Downloading a work file from a website

Editing a work file

Evaluation of student files

Output of student results and teacher comments

Edit the “Results” table

Delete a row from the “Results” table

Log out

1. Registration

a) Open the website and click on the "Register" button in the upper right corner, at the right end of the top bar.

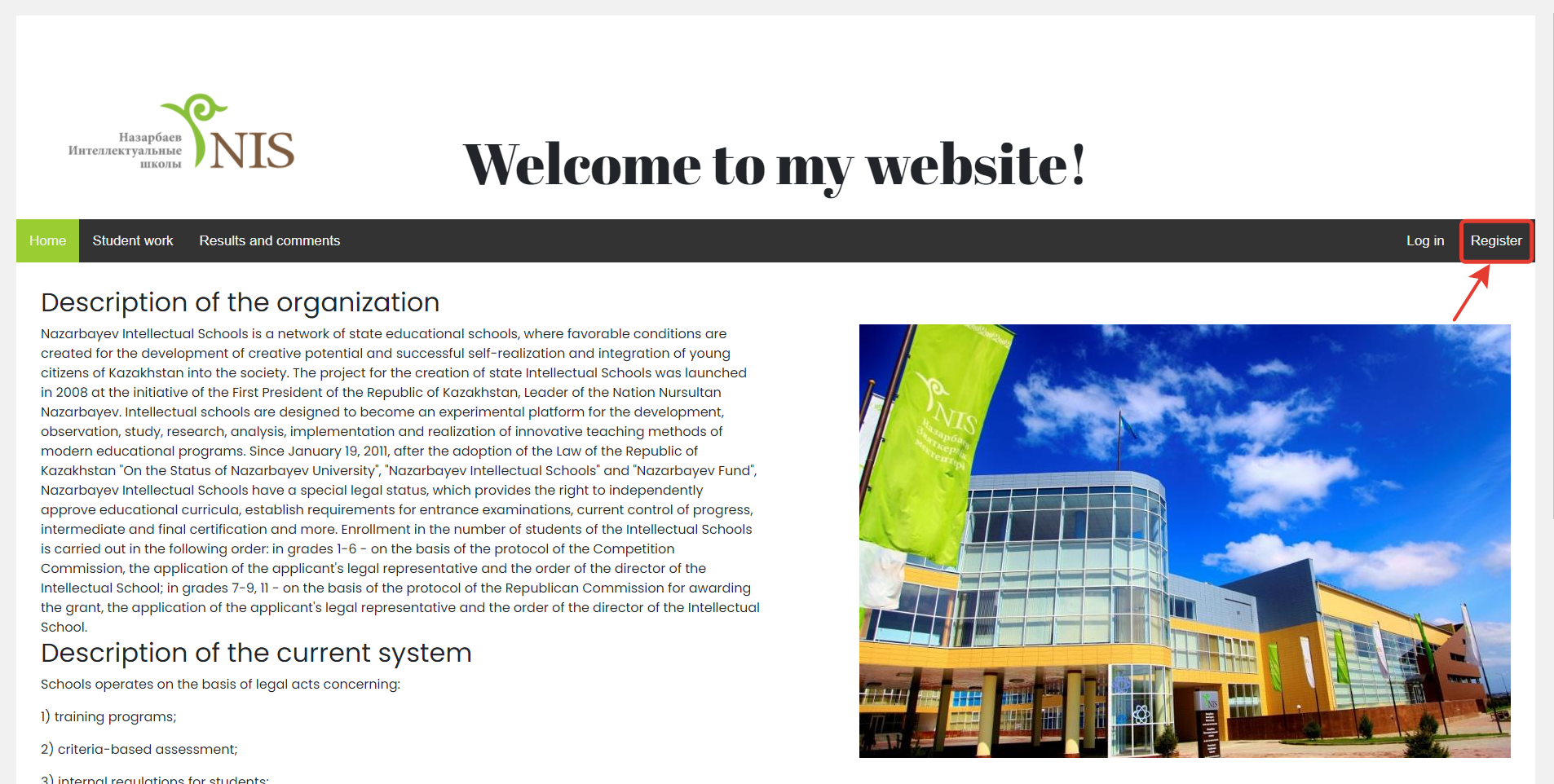


Figure 43. Register button location

b) Fill in all the input fields required to register a new user and click the "Register" button below.



Figure 44. Registration form

2. Authorization

a) Click on the "Log in" button in the upper right corner, on the right end of the top bar.

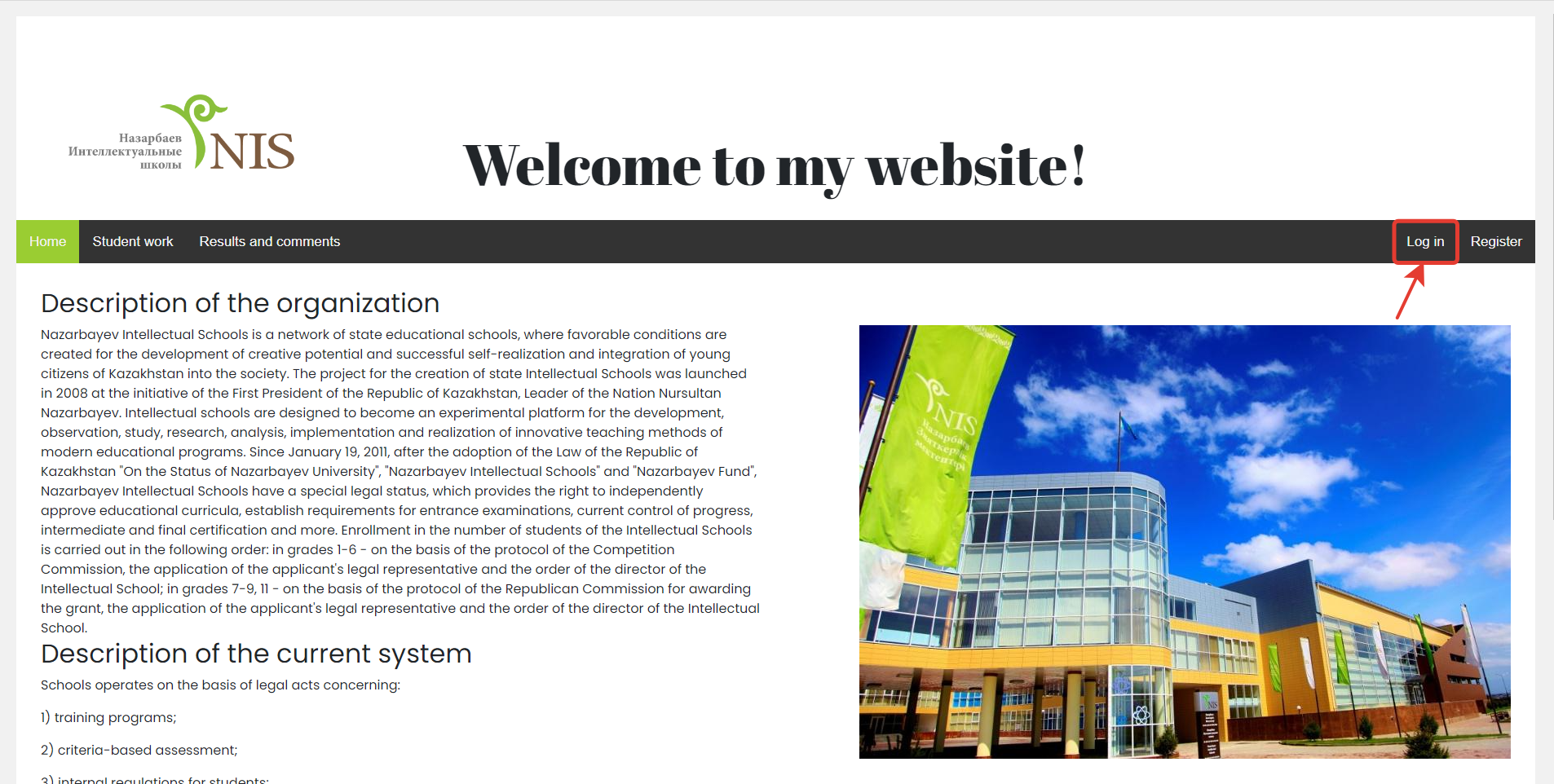


Figure 45. Log in button location

b) Fill in all the fields required to enter the system and click the "Submit" button below.

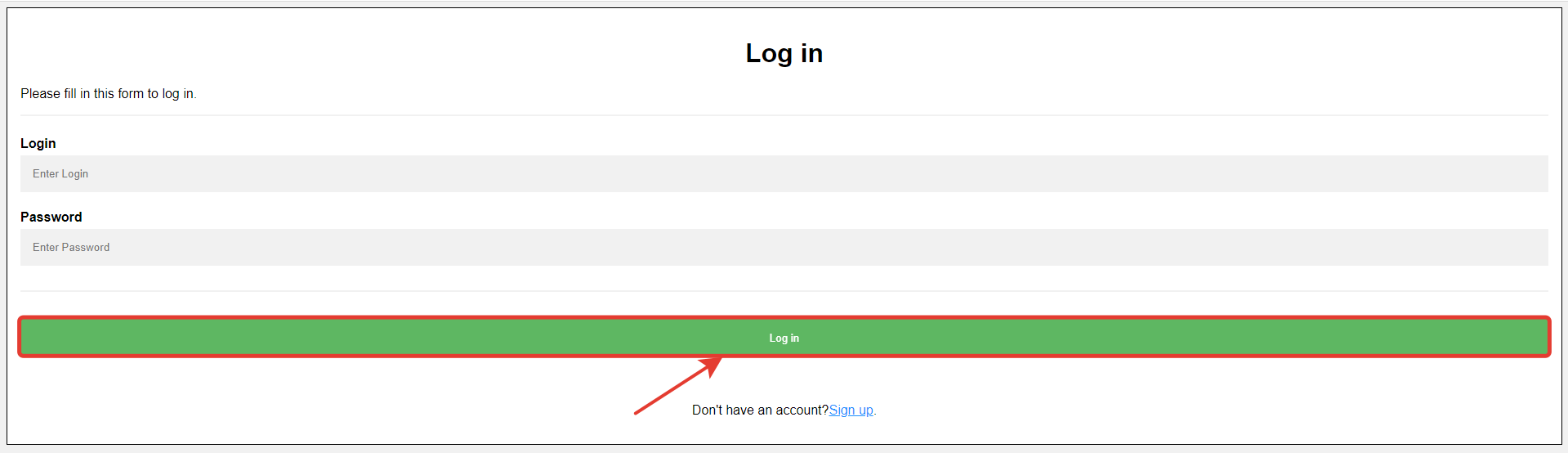


Figure 46. Log in form

3. Uploading a file to a website

a) On the “Student work” web page, click on the “Upload work” button below the top bar.

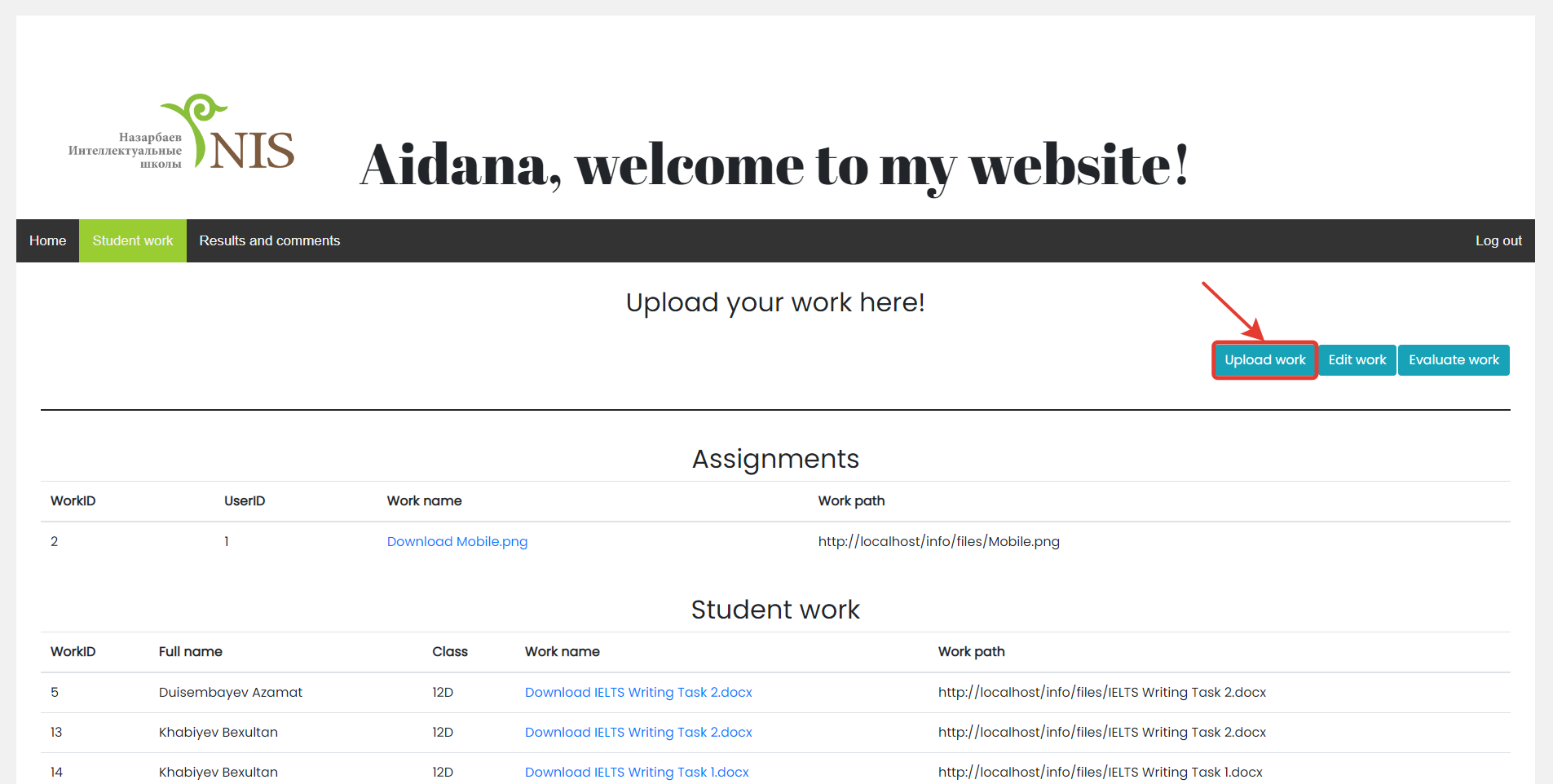


Figure 47. Upload work button location

b) In the form that opens, click on the "Browse" button to the right of the input field and select the file you need to upload.

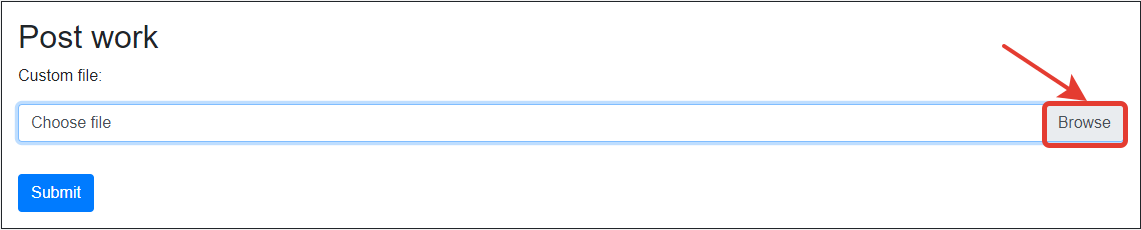


Figure 48. Upload work form

c) As soon as the file is displayed in the input field, click on the "Submit" button.

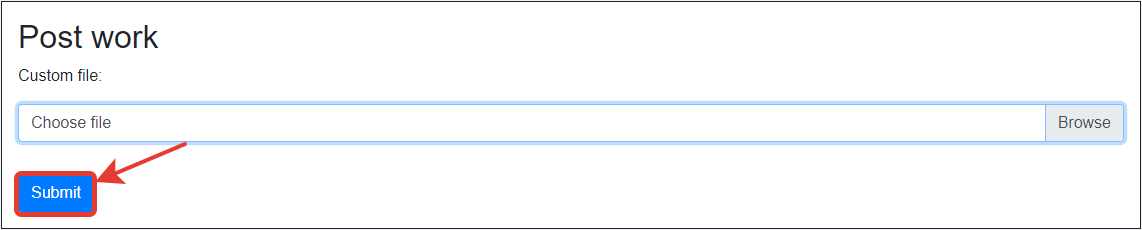


Figure 49. Submit button location

4. Student files output

a) In order to view the files uploaded by the student, click the "Student work" button and you will see a table with all the files.

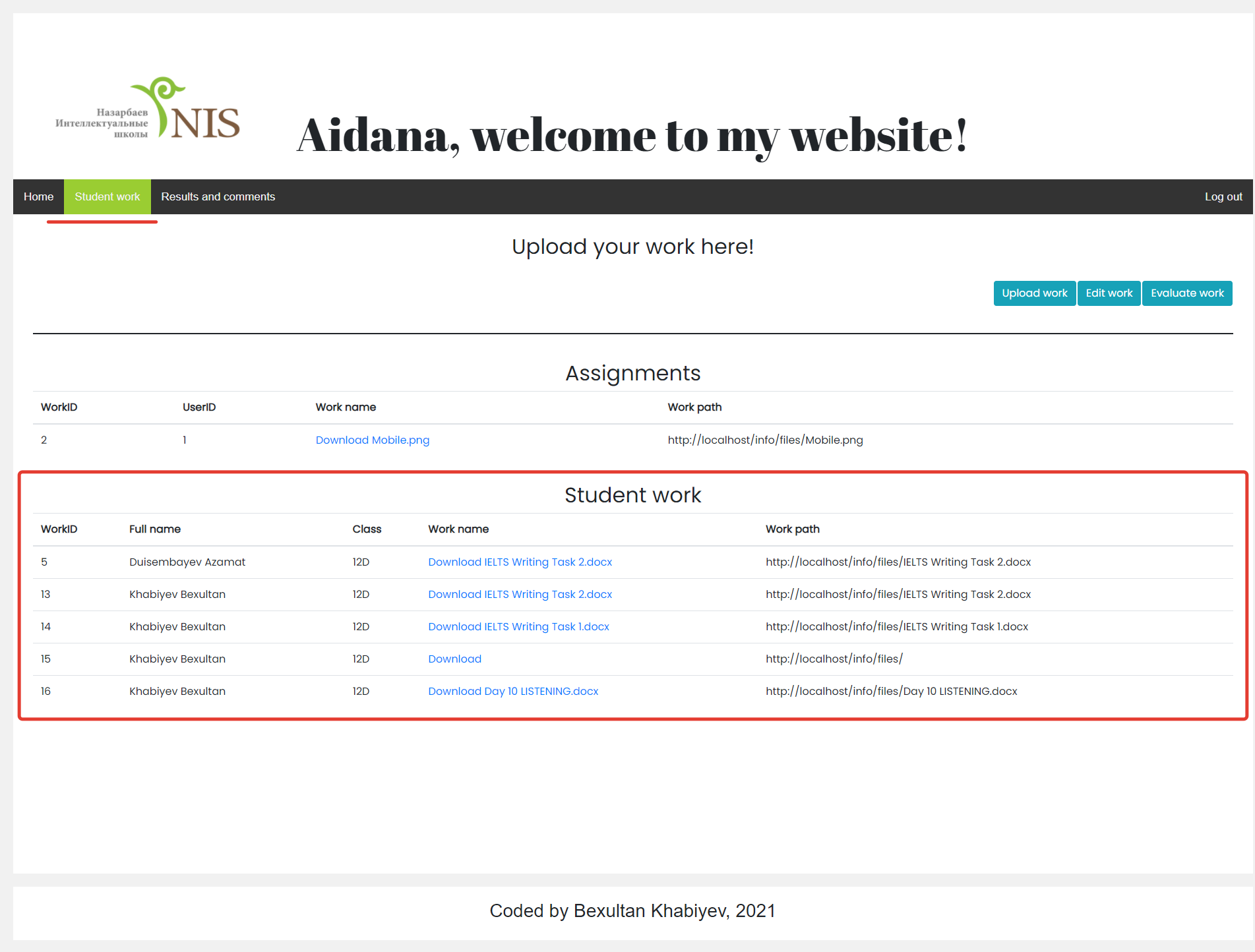


Figure 50. Student work table

5. Downloading a file from a website

a) On the web page "Student work" in the appeared table "Student work" click on the name of the file that you want to download. The file will automatically download to your computer

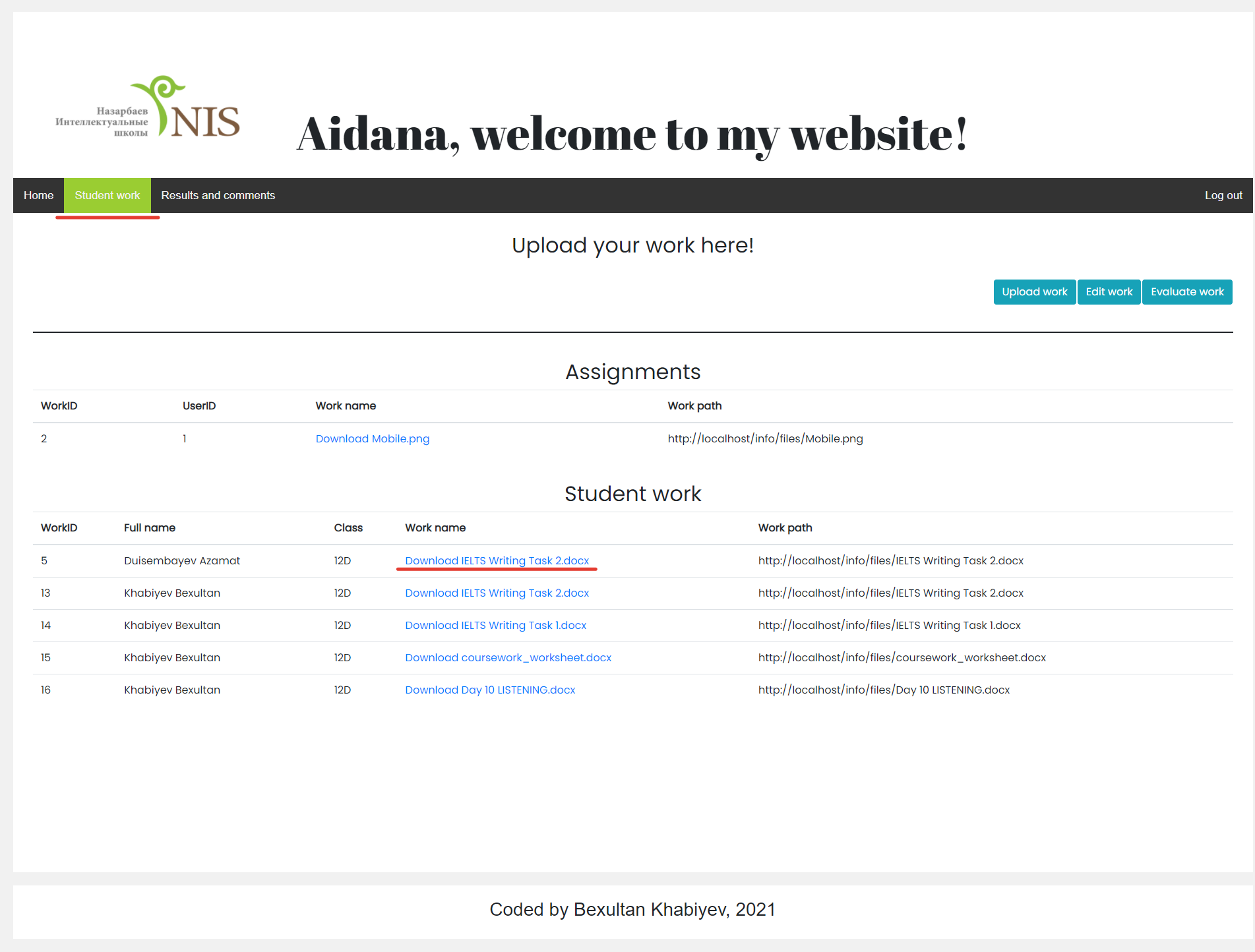


Figure 51. Link for file downloading

6. Editing a work file

a) On the “Student work” web page, click the “Edit work” button below the top bar.

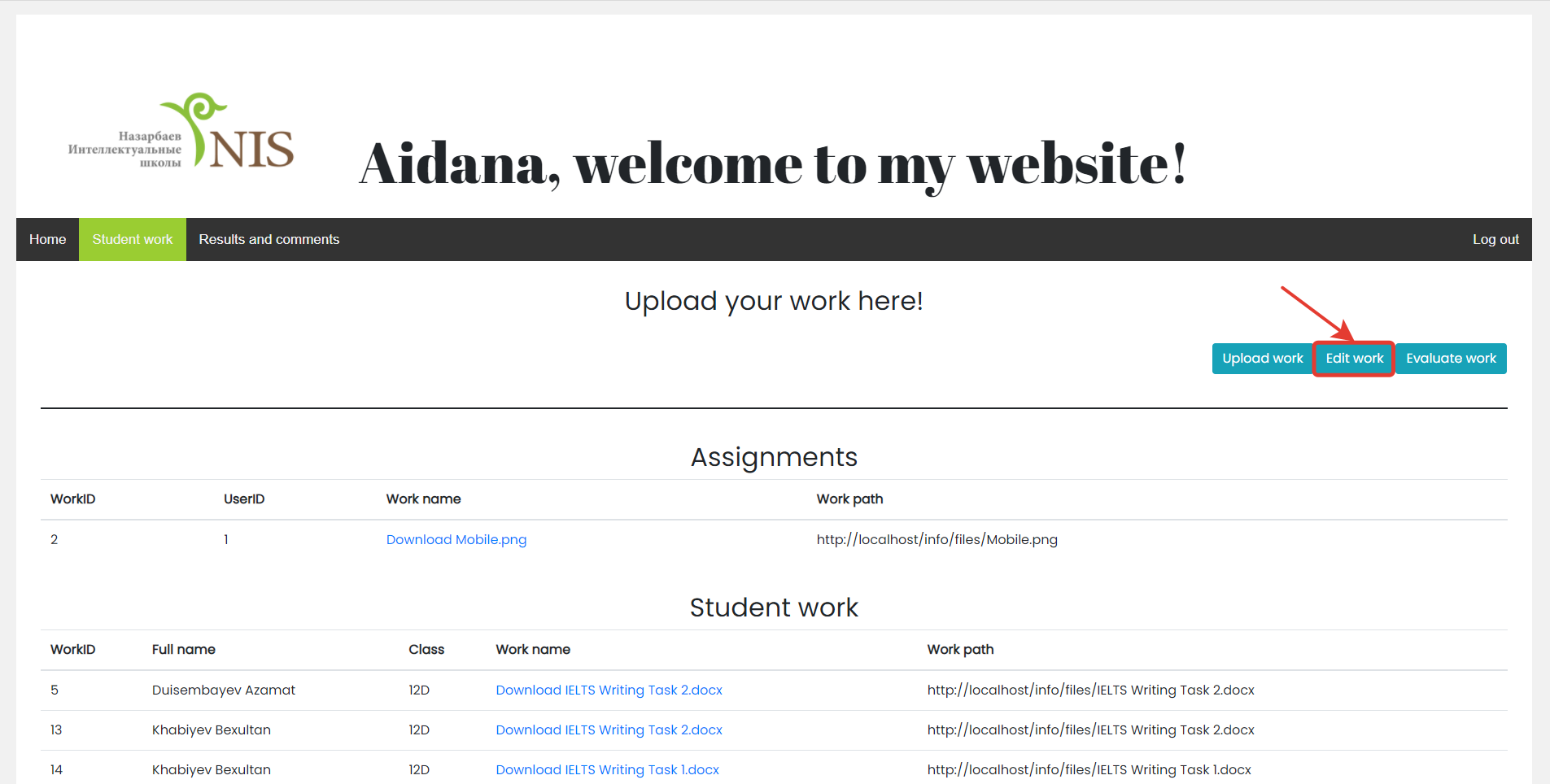


Figure 52. Edit work button location

b) Enter the ID of the work you want to edit and upload the modified or other work file.

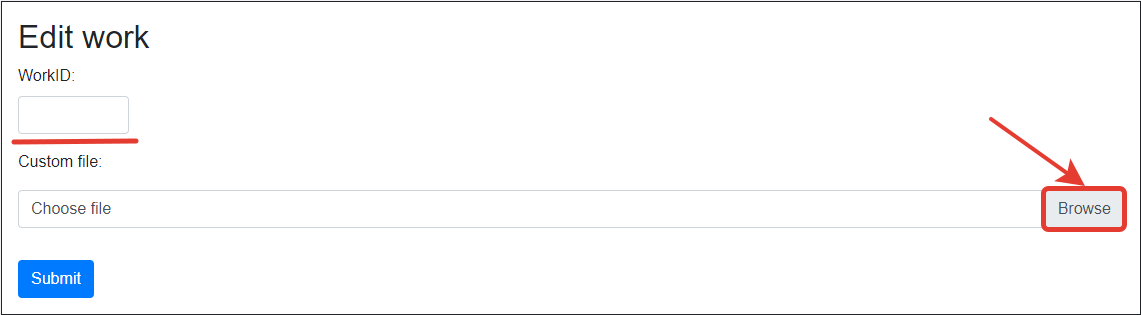


Figure 53. Edit work form

c) As soon as the uploaded file is displayed in the input field, click on the "Submit" button below.

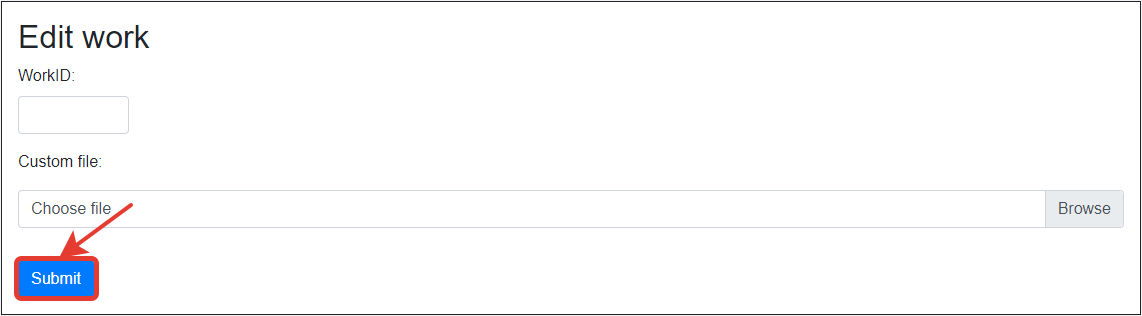


Figure 54. Submit button location

7. Evaluation of student files

a) On the "Student work" web page, after memorizing the ID of the work you want to evaluate, click on the "Evaluate work" button.

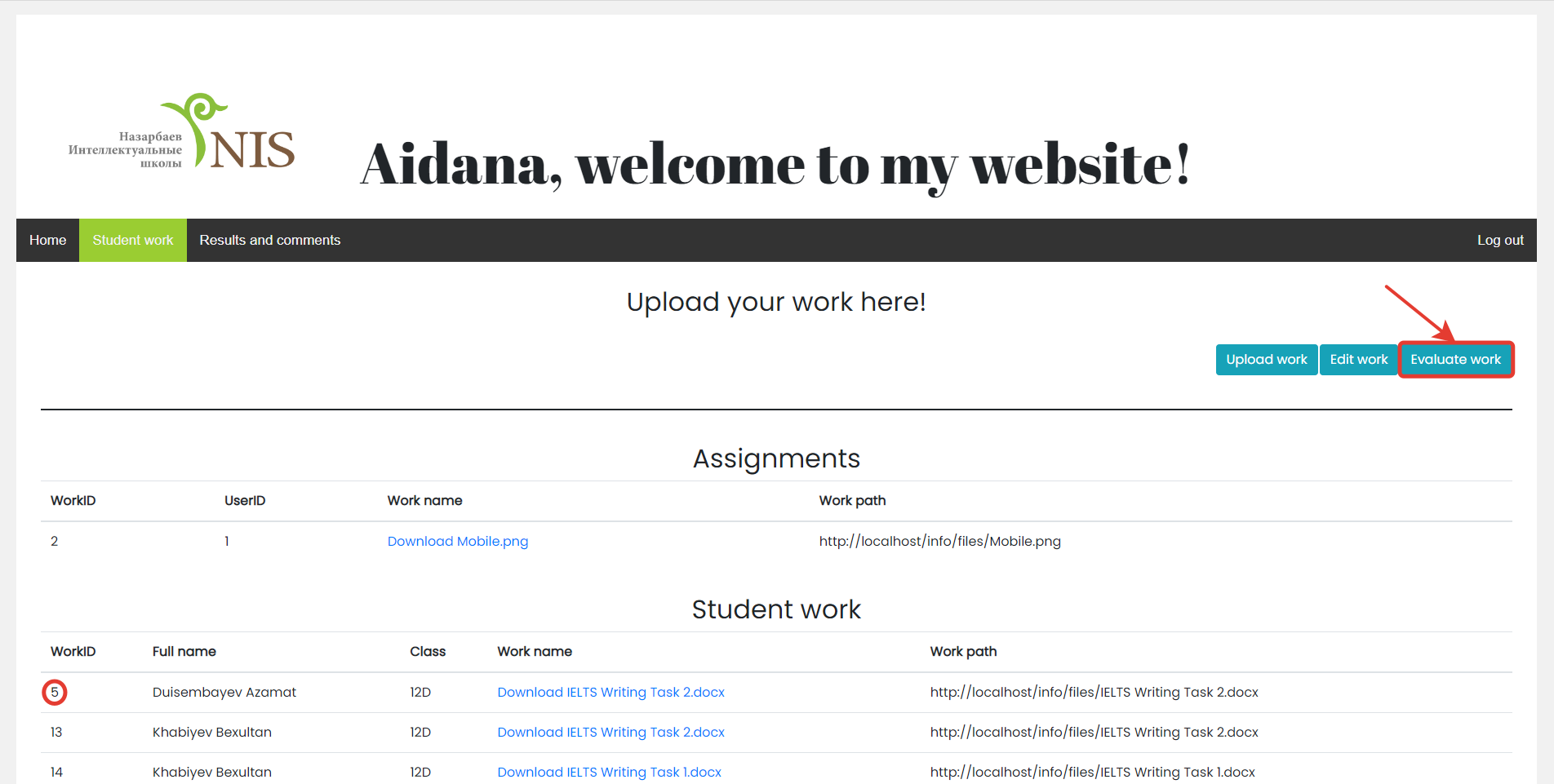


Figure 55. Evaluate work button location

b) In the form that appears, fill in all the input fields and click the "Submit" button below.



Figure 56. Submit button location

8. Output of student results and teacher comments

a) In order to view the student's results and the comments left by the teacher, you need to open the "Results and comments" web page; all the results and comments will be displayed in the table that appears.

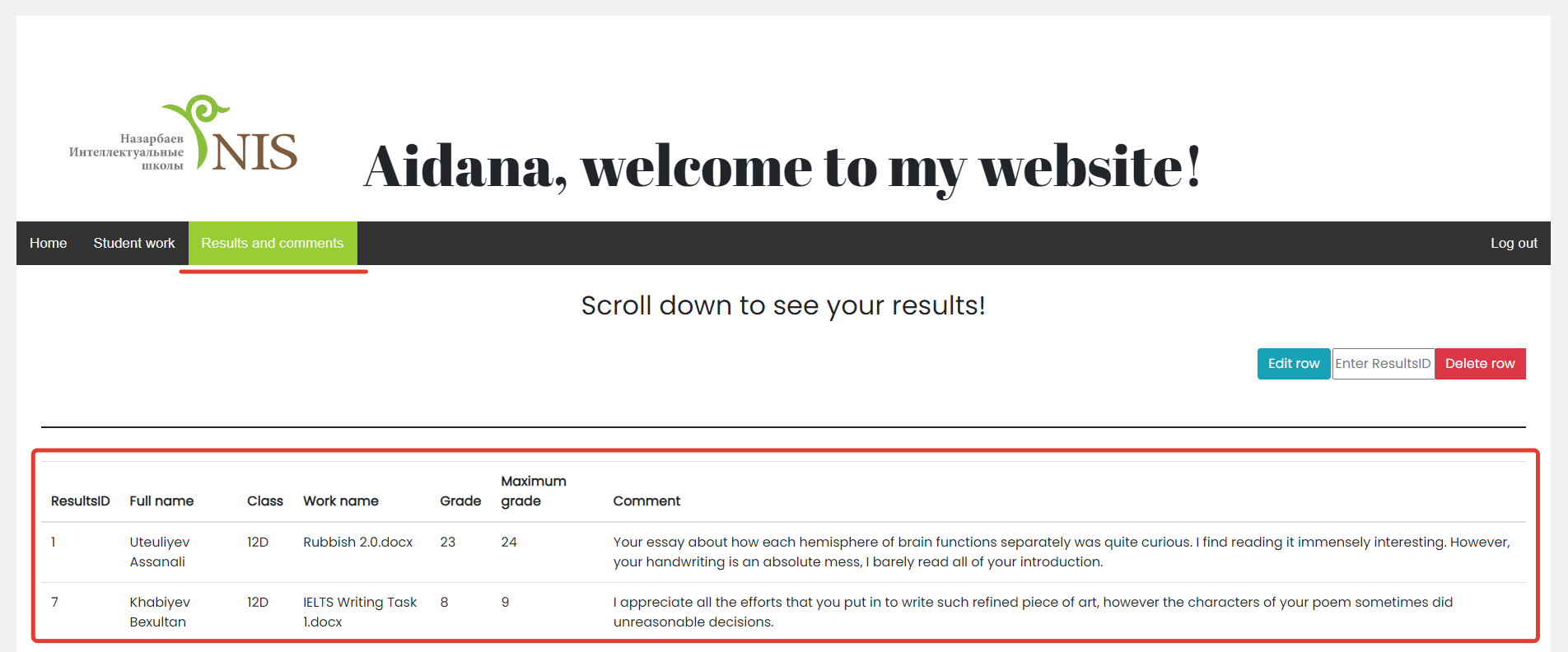


Figure 57. Results and comment table

9. Edit the “Results” table

a) To change the contents of a row from the table on the “Results and comments” web page, click the “Edit row” button below the top bar.

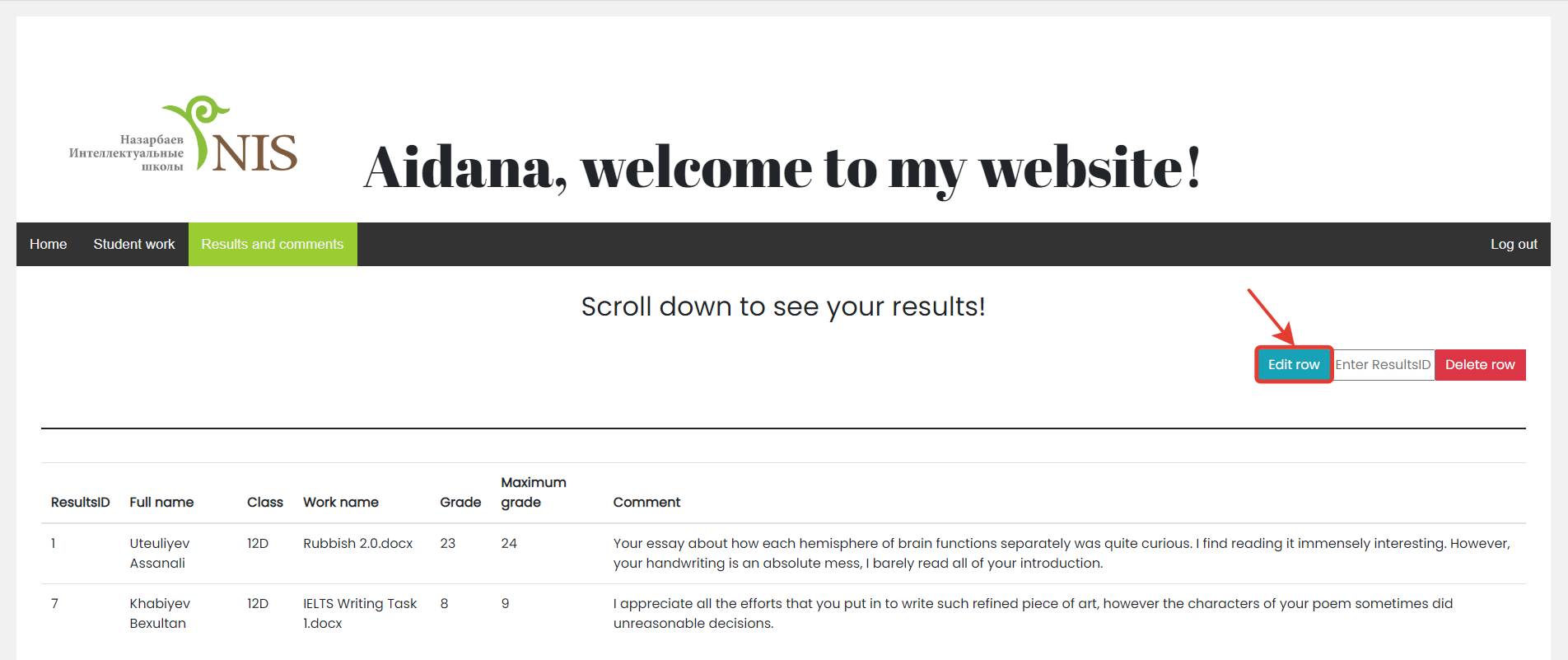


Figure 58. Edit row button location

b) In the form that opens enter the ResultsID of the row you want to edit and the rest of the required data. Then, click on the “Submit” button below.

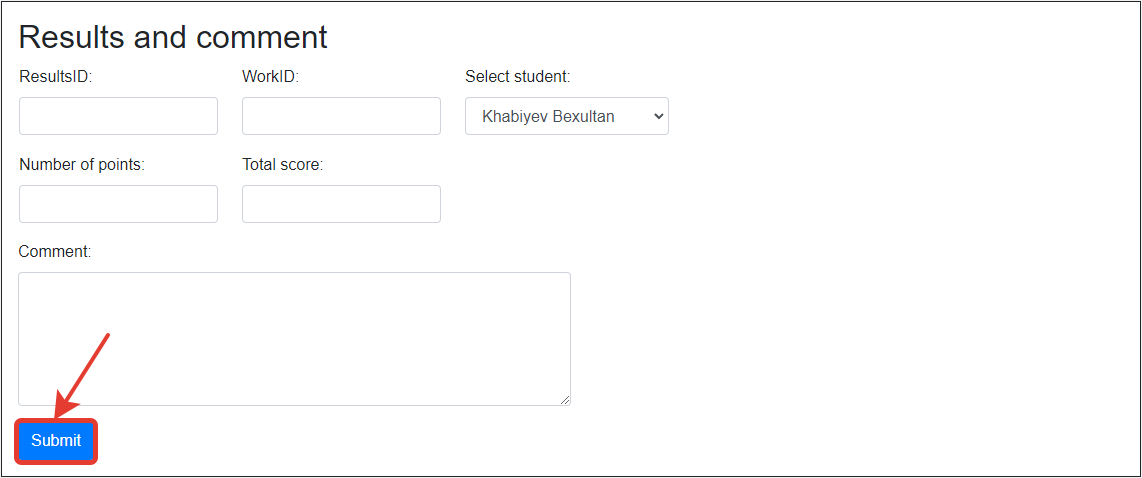


Figure 59. Results editing form

10. Delete a row from the “Results” table

a) On the "Results and comments" web page, in the input line to the left of the "Delete row" button, enter the ResultsID of the row you want to remove from the table.

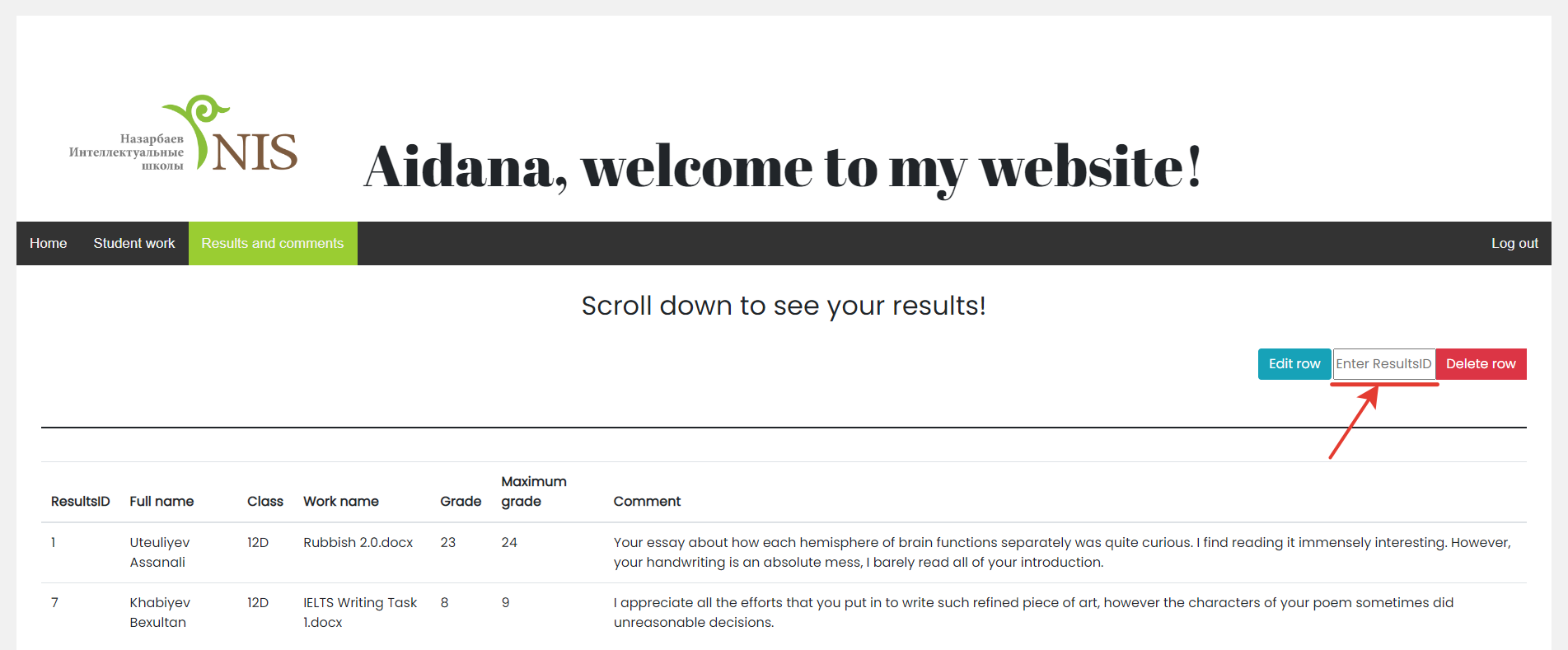


Figure 60 Enter ResultsID input line location

b) Click on the “Delete row” button.

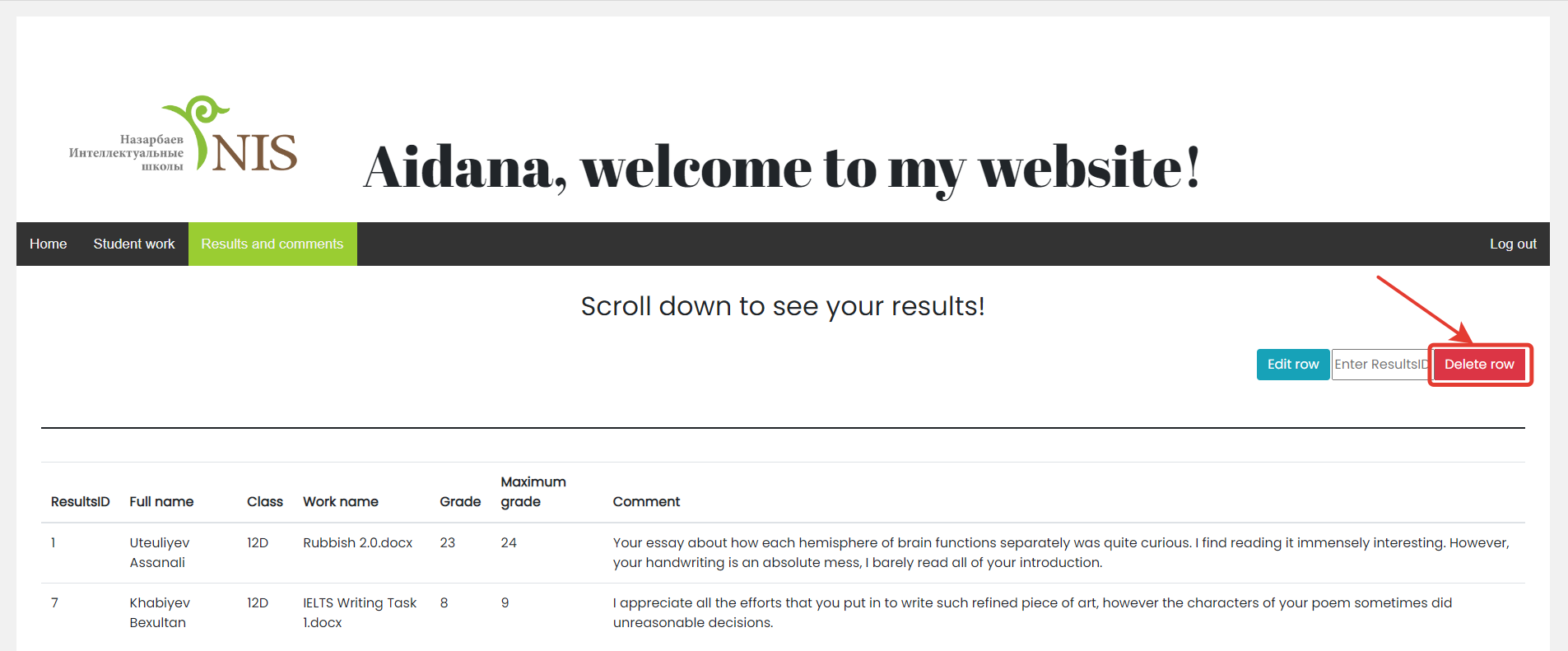


Figure 61. Delete row button location

11. Log out

a) To log out of the system, you need to click on the “Log out” button at the right end of the top bar.

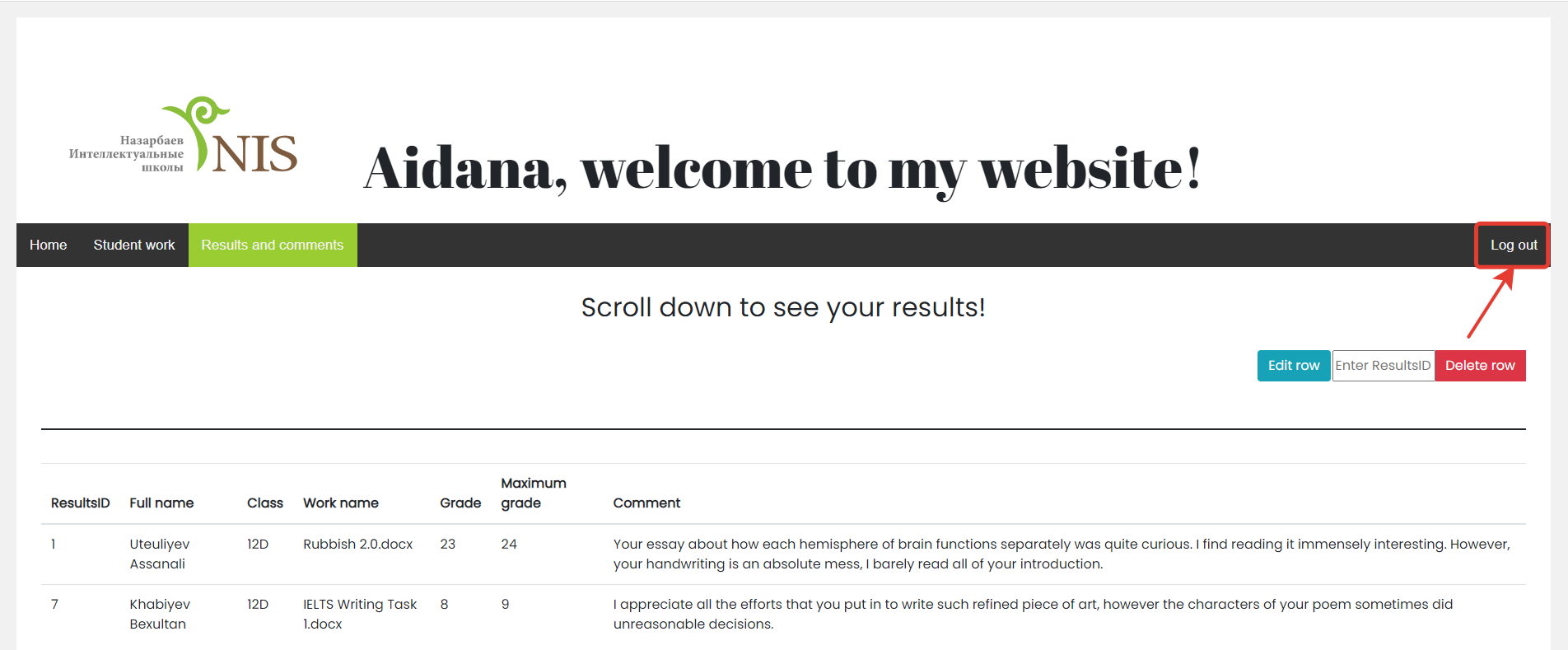


Figure 62. Log out button location

## **Guide to common mistakes**

|  |  |  |
| --- | --- | --- |
| **Error** | **Cause** | **Solution** |
| You must log in first! | The user entered the web page "Student work" or "Results and comments" without logging into the system | You must register or log in to the system to get access to these pages |
| You must fill in this field | The user left one or more input fields blank when filling out a form | Fill in all the fields required by the system |
| Your name is longer than 30 characters or shorter than 2 characters! | The length of the entered surname is longer or less than the possible number of characters | Write the surname within the possible number of characters |
| Your surname is longer than 30 characters or shorter than 2 characters! | The length of the entered name is not in the allowed number of characters | Enter name according to length limits |
| Your login is longer than 30 characters or shorter than 8 characters! | The length of the login exceeds or is below the allowed limit | Make the login fit in the length limits |
| Your password is longer than 30 characters or shorter than 8 characters! | The password does not meet the security requirements of the website | Make the password less predictable by increasing the length or decrease the number of characters so that it fits in the length limits |
| You must enter your class name! | The user selected the "Student" status and did not fill in the "Class" field | Fill in the "Class" field |
| Your class name is longer than 3 characters! | The entered class name is longer than the length limit | Write the name of the class so that it does not exceed the length limit |
| Your login already exists! | The user entered a login that already exists in the system database | Come up with another unique login |
| Passwords do not match! | The user incorrectly repeated the password that he entered earlier when filling out the registration form | Check the first password and enter an identical password in the "Repeat password" input |
| Your login or password was entered incorrectly! | User entered incorrect password or login | Check the entered data for errors and write the password and login that were entered during registration |
| You have not uploaded your file! | The user did not load the file to the form | Load the required file |
| You are trying to edit another student’s work! | The user entered a workID of a work that he did not upload | Enter the workID of the work that belongs to you |
| You have written a wrong full name or workID! | The teacher attributed workID to the student, who did not upload this work | Go back to the table with all students' works to check the workID and the name of the student the teacher wants to assess and re-enter this data in the evaluation form |
| Your number of points is more than the three-digit number! | The number of points has exceeded the possible number | Enter a lower number of points |
| Your total score is more than the three-digit number! | The total score is more than the acceptable number | Decrease the total score |
| Your comment is longer than 250 characters! | The teacher's comment was too long | Make the comment shorter |
| You have written a wrong full name, workID or resultsID! | The teacher, when editing the row of the "Results" table, specified an incorrect resultsID, or selected an inappropriate student name and work ID | Return to the web page with the "Results" table, check if you entered all the data correctly and enter them again |

**Backup procedures**

To ensure data safety in the system, it is necessary to carry out the backup procedure once a month. Keeping a copy of the data will prevent the complete loss of information if the system has problems or becomes unstable. A file with a copy on another storage will always be available.

* + - 1. To export the database open XAMPP control panel and then click the "Start" button for the “Apache” and “MySQL” modules. After the modules have started working, click the "Admin" button opposite to “MySQL”.

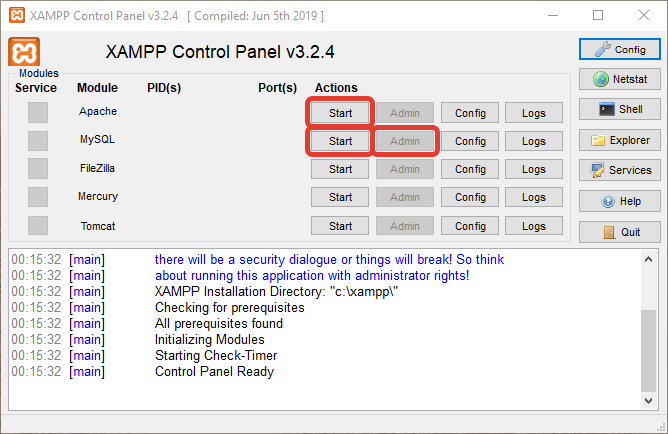


Figure 63. XAMPP control panel

* + - 1. The “phpmyadmin” web page will open in the default browser, on which you need to select the "project" database.

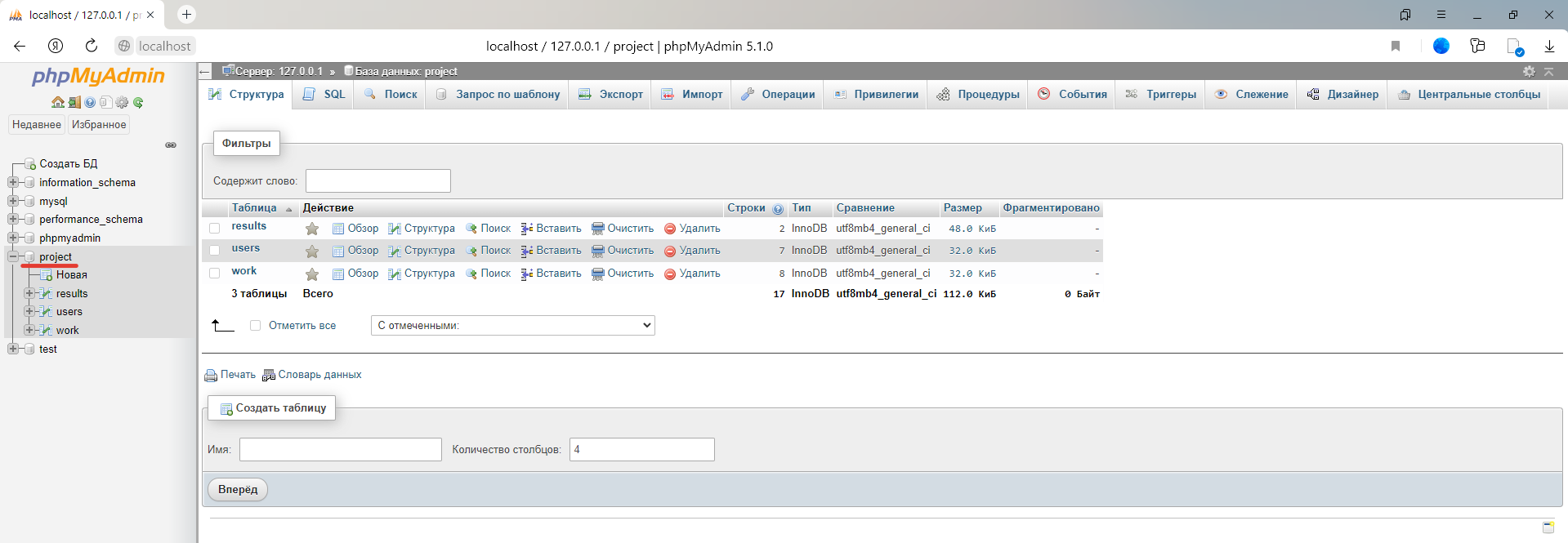


Figure 64. Project database location

* + - 1. Next, click “Экспорт” on the top bar of the web page and select "Обычный – отображать все возможные настройки" as the export method.

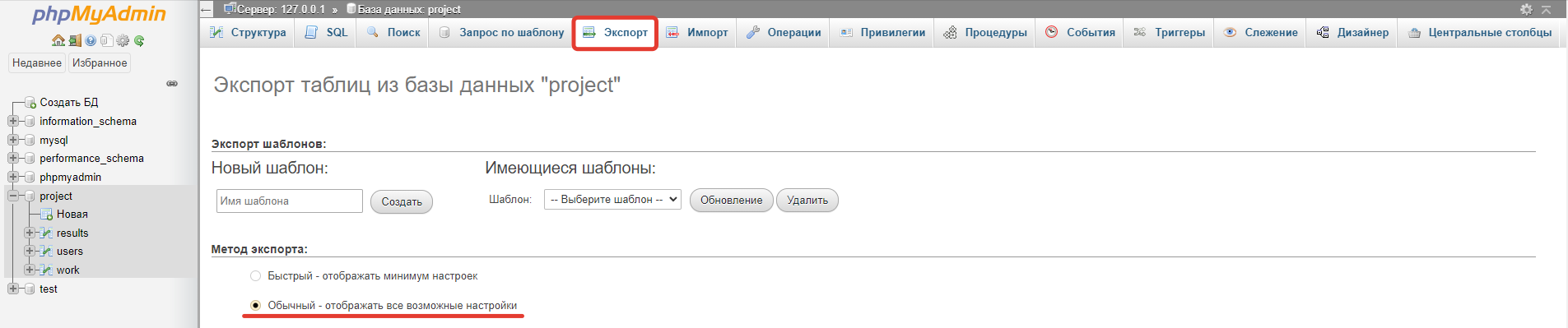


Figure 65. Needed export method

* + - 1. Scroll down to “Параметры создания объектов” and tick the “Добавить выражение CREATE DATABASE / USE” and “Добавить выражение DROP TABLE / VIEW / PROCEDURE / FUNCTION / EVENT / TRIGGER”.



Figure 66. Required parameters for creating objects

* + - 1. Now scroll to the bottom of the web page and click the “Вперёд” button in the lower right corner and the database will be downloaded to your computer.

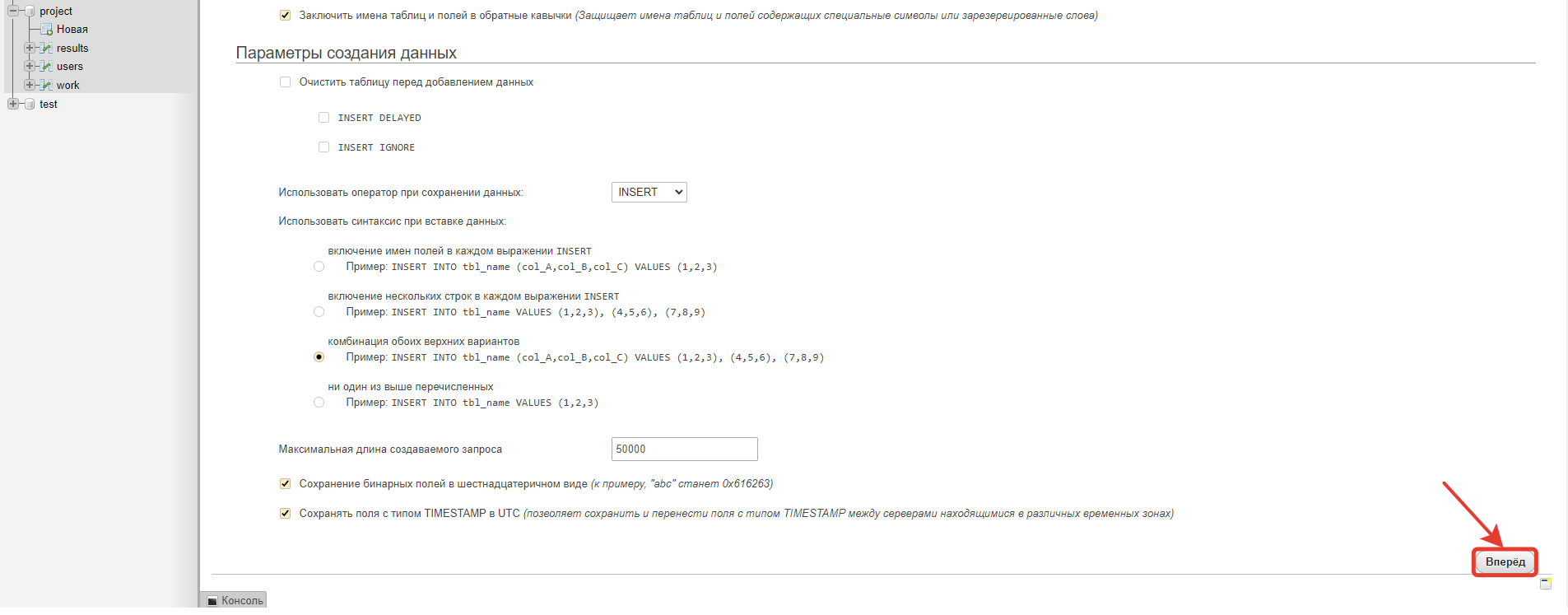


Figure 67. Submit button location

* + - 1. Transfer a file with a database to another storage (cloud storage or flash drive)

**Glossary**

**Backup** isa copy of a file, etc., that can be used if the original is lost or damaged.[[1]](#footnote-1)

**Database** is an organized set of data that is stored in a computer and can be looked at and used in various ways.[[2]](#footnote-2)

**Form** is a graphical interface that is used to collect data from users.

**Top bar** is a graphical control element at the top of the screen that gives access to all the functions of the website.

**Web page** is a document that is connected to the World Wide Web and that anyone with an internet connection can see, usually forming part of a website.[[3]](#footnote-3)

**Index**

button 94, 95, 96, 97, 98, 99, 100, 101, 105, 106

input 95, 98, 99, 100, 103

login 103

password 103

row 93, 99, 100, 101

system 94, 101, 103, 105

table 96, 99, 100, 104

web page 103, 105, 106

website 95, 96, 103, 107

# Evaluation

## Discussion of the degree of success in meeting the original objectives

At the stage of evaluating the work, it is necessary to compare the website’s functions with the goals set before implementing the project. This allows me to ensure that the new system meets the client's requirements. Moreover, I have added several additional features that made the website even more functional than planned.

1. Provide a friendly interface for users
2. Provide registration of new users
3. Implement a log in process
4. Add a function for users to upload files
5. Add a function for the teacher to leave comments and set results for summative assessment
6. Provide an opportunity to view student files
7. Provide an opportunity to view results for summative assessment

|  |  |  |
| --- | --- | --- |
| **№** | **Evaluation** | **Evidence** |
| 1 | The website has a friendly and intuitive interface that allows a new user to navigate easily. During the creation of the website, the Nazarbayev Intellectual School symbol and photographs were used to indicate the connection of the new system with this organization. The website’s color palette mainly consists of not very bright colors (black, white, gray, turquoise) so that the interface does not strain the eyes. When an authorized user logs into the website, the interface greets him by name so that the user knows they have successfully logged into the system. | Testing  Page: 86 |
| 2 | The new system has new user registrations, which makes it possible to grade students individually. The registration process satisfies the basic principles of data security, as it does not allow new users to register with the same logins or enter inappropriate passwords the next time. Length validation determines the required range of the number of characters, due to which short passwords are not allowed, which makes them less predictable, as well as too long passwords that the user can forget. | Testing  Page: 67 |
| 3 | User authorization has been successfully added to the website. In the authorization process, as in registration, sessions are used, which allow you to store general data about the user while he is on the website. During the implementation of this function, there were no noticeable difficulties, since we learned this in computer science lessons. | Testing  Page: 74 |
| 4 | On the website, an authorized user is given the opportunity to upload work files to the database in order for them to be assessed by the teacher. Adding this function was a challenge for me since working with files is not explained in the school course. However, thanks to the help of the huge community of programmers on social media, I was able to find a solution | Testing  Page: 76 |
| 5 | On the website, the teacher (admin) has the ability to post results and write comments on the work of students. The most difficult task was to prevent the situation when the teacher assessed the wrong job that he wanted. For this reason, the teacher must enter the ID of the work and the name of the student he wants to assess. If the entered ID of the work does not belong to the student that the teacher wants to evaluate, then the program will generate an error. | Testing  Page: 78 |
| 6 | In order to present information on the website effectively, a separate page has been created where users can see the uploaded work files. This allows the teacher to understand which students have uploaded their work. The most difficult part of the implementation process was the use of loops to create tables. Only with the help of the graduates of our school, I was able to understand how data can be displayed using loops. | Testing  Page: 85 |
| 7 | On the "Results and comments" web page, the output of the students' results and the teacher's comments in the form of a table was implemented. Thus, students can see how well they completed the assignments. The main problem was to present data from two different tables as one. It took me a lot of time on the internet forums to realize that I need to use the “JOIN” command to combine the data from the two tables. | Testing  Page: 86 |

## Evaluate the client’s and user’s response to the system

Below there is an email with a customer evaluating the new system. The client tested the website for a week and shared his own opinion about it.

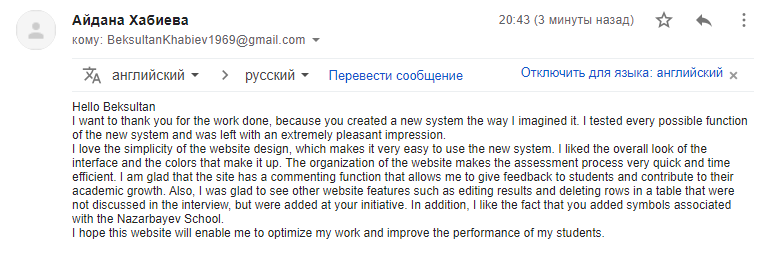


Figure 68. Client evaluation of the new system

In conclusion, the feedback provided by my client indicates that the new system has achieved all the goals set at the design stage and fully meets the client's expectations. With my client's approval, during the implementation process, several features were added that significantly improved the new system. The client appreciated my design skills and claims that the interface was done as he imagined it.

Firstly, in the new system for new users there is an opportunity to register and for existing users to log in. There were no significant problems with the implementation of the registration and authorization function, since I was instructed on how to do this in computer science lessons. All students were provided with ready-made examples of program code, on the basis of which these functions were implemented. The main difficulties were associated with the output of data to the website, as it involves the use of loops and special SQL commands. The client was very pleased that it was possible on the website to give feedback to students regarding the work performed. He believes that this feature will allow students to improve their academic level and greatly facilitate teachers' work.

Secondly, the client notes that he was pleased with the overall automation of the entire assessment process. The new system was able to optimize this process in such a way that it goes much faster. Throughout the implementation phase, I eliminated all possible problems that could make the new system inconvenient or ineffective. As a result, the functions for editing the working file and the table of results were added to fix errors and inaccuracies that the user can make. In addition, on the "Results and comments" web page, a function for deleting rows has been added, which can be applied in case of the accumulation of an abundant amount of unnecessary information. All of these measures have been taken to ensure the new system is multifunctional and easy to use.

Thirdly, in his letter, my client highlighted the interface of the website. She believes that the simplicity of the interface makes the process of using the new system extremely understandable and comfortable. The client also highly appreciated the appearance of the website. During the creation of the website's appearance, such colors (green, black, turquoise, and gray) were chosen that will be pleasing to the eye and will not cause irritation. In the main interface of the website, the symbol of the Nazarbayev Intellectual School is highlighted, which indicates the direct connection of the new system with this network of educational schools. The highlighted tabs on the top bar make it easy for the user to navigate, and the logged-in username greeting notifies him that he has successfully logged in. These small details make the new system very convenient and user-friendly.

Moreover, it is worth noting that I spent most of the time trying to implement interactions with files since I did not have enough experience. The main questions were how to implement adding work files to the database of the new system and how to make sure that the teacher (admin) could download the files uploaded by the students. After a huge number of hours spent on thematic Internet forums and discussions with alumni of our school, I found out the answers to my questions. I used the global variable FILES, which allowed me to upload files from the form to the system, and I also provided the filenames as download links for the teacher to view.

Finally, in my opinion, I achieved all my goals and even did more than planned. A client of mine shared extremely positive feedback on the new system and expressed his complete satisfaction. I have put a tremendous amount of effort into making the website user-friendly and feature-rich. I find the results of my work extremely impressive since I had never written a program in PHP before developing this project. I think creating a website is a rewarding experience for me and I believe that it will be of benefit to me in the future.

1. https://www.oxfordlearnersdictionaries.com/definition/english/backup?q=backup [↑](#footnote-ref-1)
2. https://www.oxfordlearnersdictionaries.com/definition/english/database?q=database [↑](#footnote-ref-2)
3. https://www.oxfordlearnersdictionaries.com/definition/english/web-page?q=web+page [↑](#footnote-ref-3)