**Exam Management System**

User Manual

Students: Roi Tiefenbrunn, Ofek Nov, Idan Aharoni, Mor Abo

Customer: Dr. Mayer Goldberg

Contents

[Introduction 3](#_Toc171356360)

[Training 4](#_Toc171356361)

[Use Cases 4](#_Toc171356362)

[1. Add User 4](#_Toc171356363)

[2. Creating a Task 4](#_Toc171356364)

[3. Manage Questions 5](#_Toc171356365)

[4. Generate Exams 5](#_Toc171356366)

[5. Finishing a Task 6](#_Toc171356367)

[Reference 7](#_Toc171356368)

[User Facing Functions 7](#_Toc171356369)

[Limitations 11](#_Toc171356370)

[Index 11](#_Toc171356371)

# Introduction

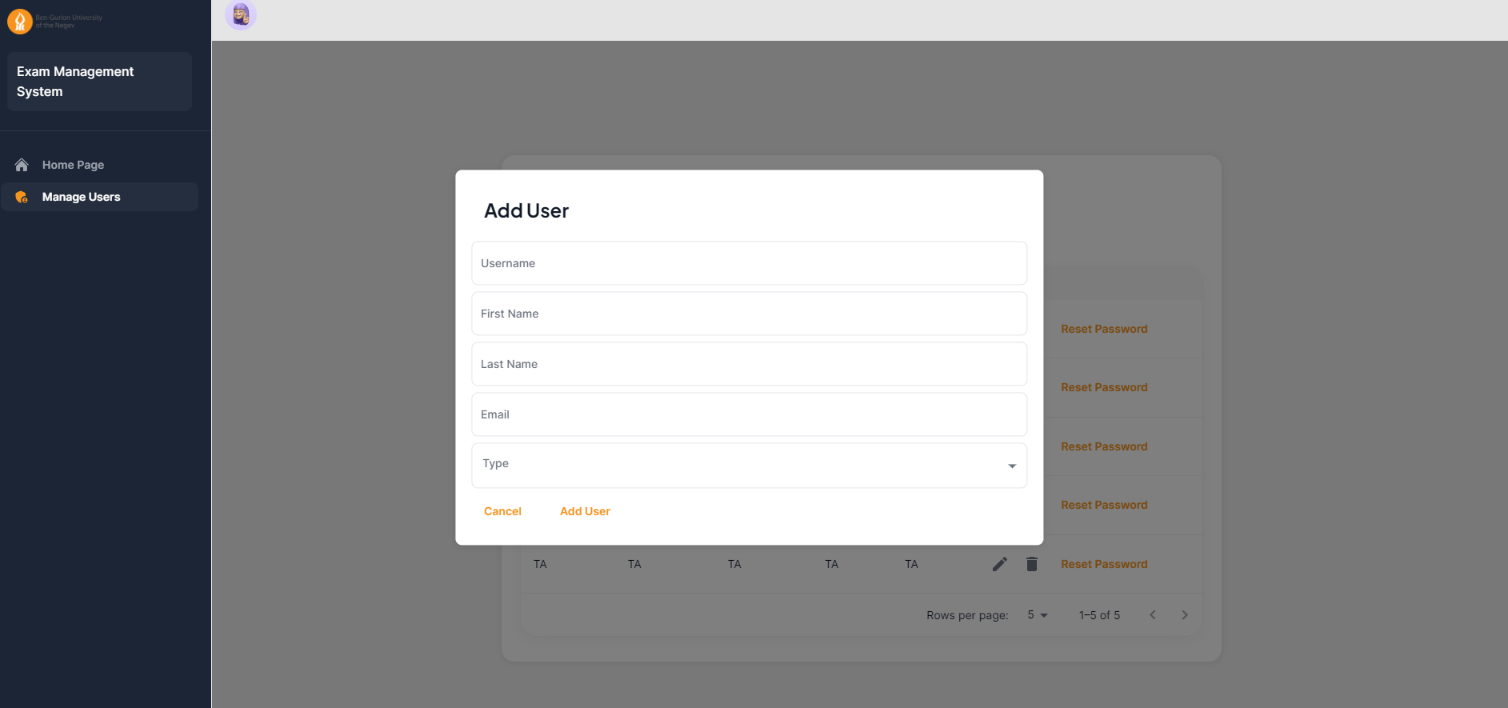
The Exam Management System (EMS) is designed to streamline the process of creating, managing, and generating exams. It offers a range of functionalities to help administrators, lecturers, and TAs manage their tasks efficiently. This manual serves as both a training guide and a reference document, providing detailed instructions on how to use the EMS to accomplish various tasks.

# Training

## Use Cases

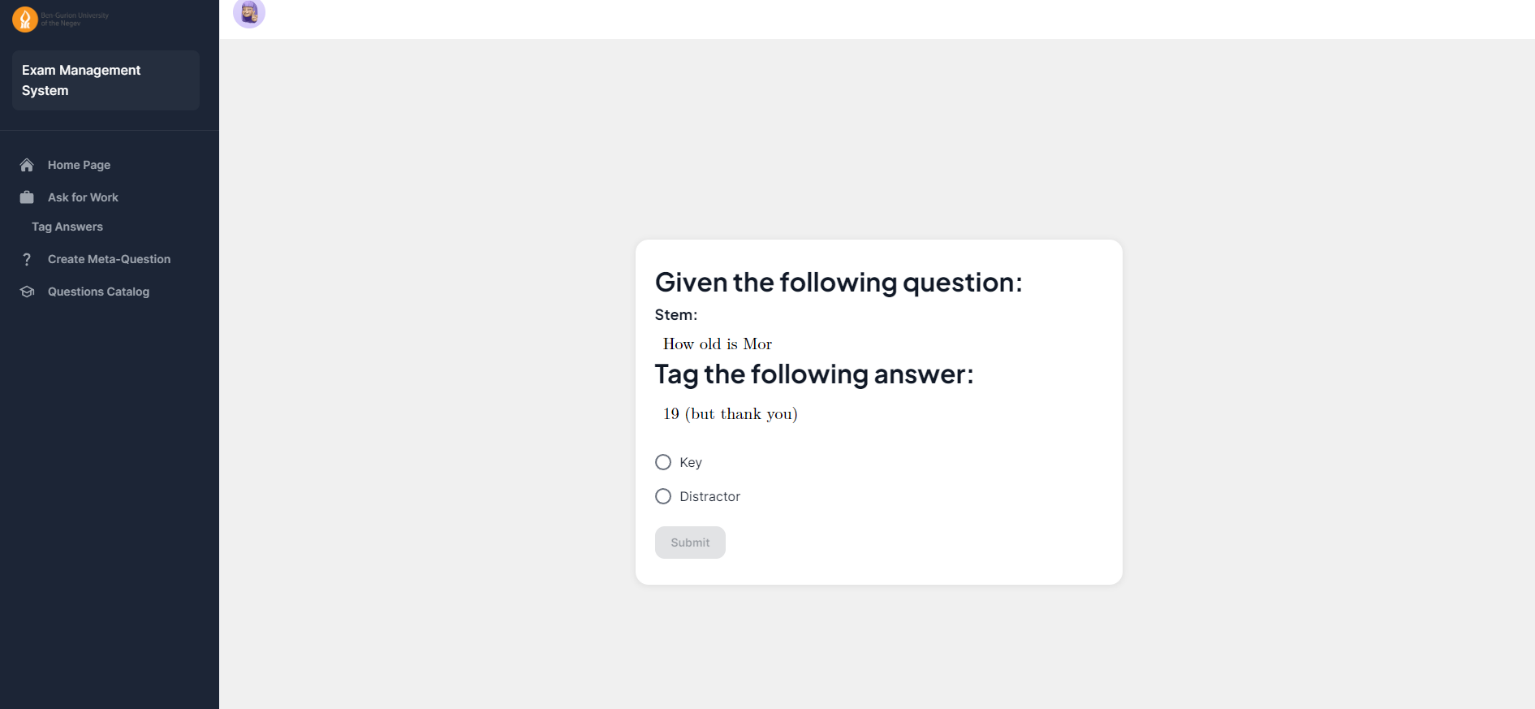
### 1. Add User

* **Description:** Allows administrators to add new users to the system.
* **Actor:** Administrator
* **Preconditions:** The user must be logged in as an administrator.
* **Basic Flow:**
  1. The administrator accesses the course management section of the system.
  2. The administrator views a list of existing users.
  3. The administrator adds a new user to the course.
* **Alternate Flows:**
  1. If the administrator tries to add a user with an already existing username, a proper error message should be shown.



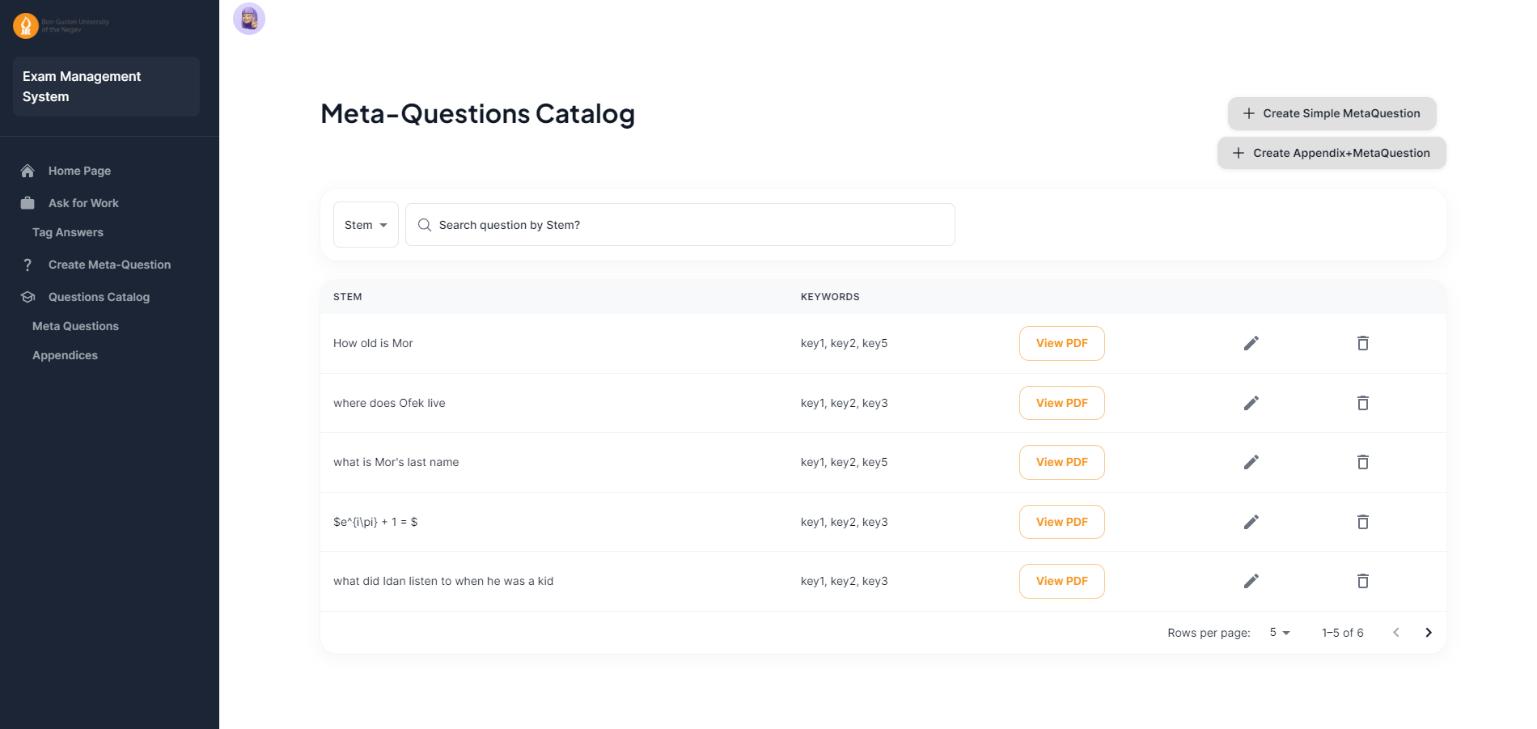
### 2. Creating a Task

* **Description:** Enables users to review questions and create tasks.
* **Actor:** Lecturer, TA
* **Preconditions:** The user must be logged in with appropriate permissions.
* **Basic Flow:**
  1. The TA accesses the review questions section.
  2. The TA submits a different answer for the question that he got.
  3. The system creates a review question task.
* **Alternate Flows:**
  1. If the TA does not make changes to the question, then a task is not created.



### 3. Manage Questions

* **Description:** Allows users to manage questions within the system.
* **Actor:** Lecturer, TA
* **Preconditions:** The user must be logged in as a Lecturer or TA.
* **Basic Flow:**
  1. The user accesses the questions management section of a course.
  2. The user creates new content such as questions, stems, meta-questions, and appendices organized by subjects and keywords.
  3. The user edits, deletes, or validates existing content as needed.
* **Alternate Flows:**
  1. Other users on the course need to review and provide feedback on proposed content changes before finalizing them.



### 4. Generate Exams

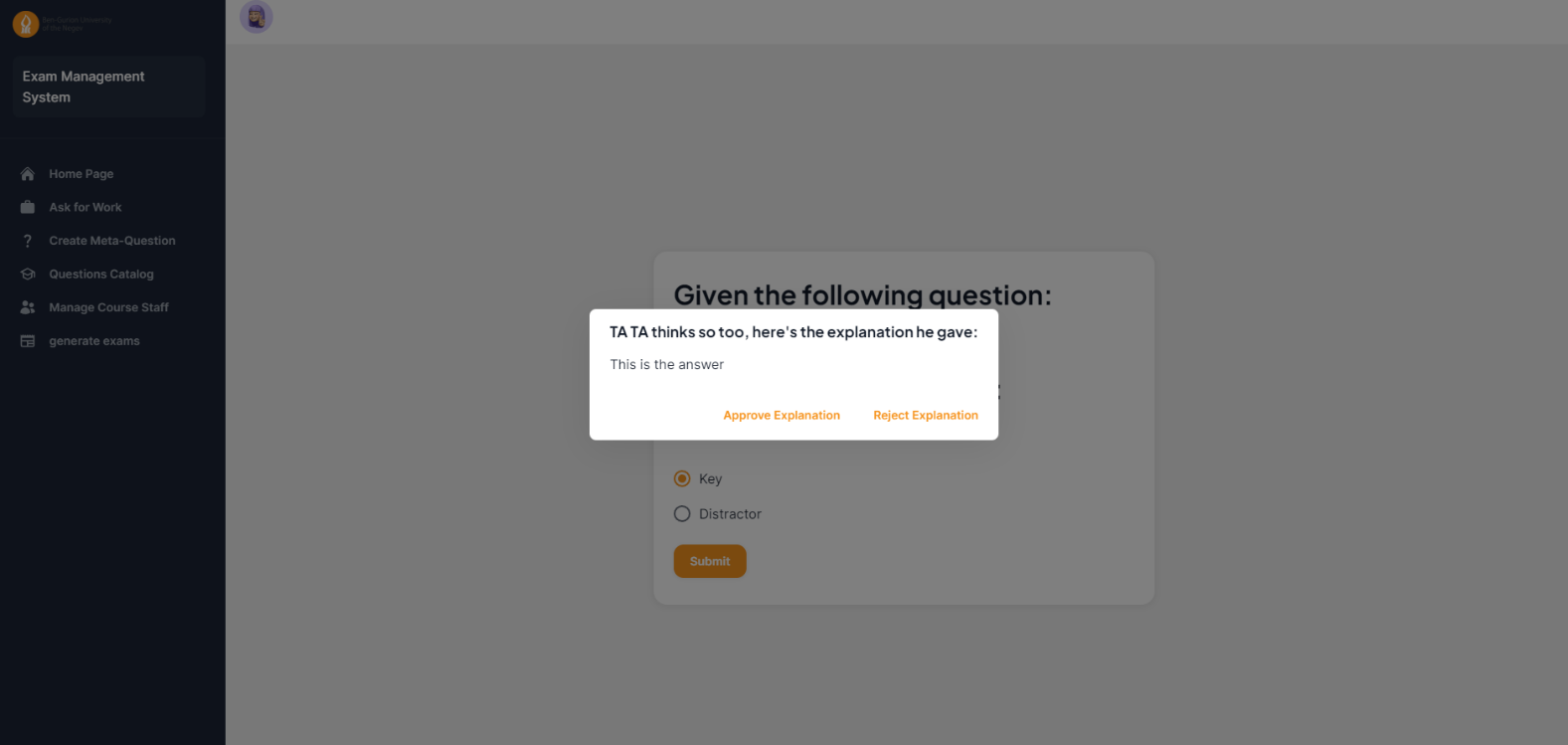
* **Description:** Allows users to generate exams based on the available content.
* **Actor:** Lecturer
* **Preconditions:** The user must be logged in as a Lecturer, and the course must have defined exam properties such as subjects, number of questions, and grading system.
* **Basic Flow:**
  1. The user accesses the exam creation section of the system.
  2. The user selects to create an exam and provides necessary details.
  3. The system generates the exam using LaTeX-based templates and exports it to a PDF file.
* **Alternate Flows:**
  1. If the user wants to preview the exam before finalizing it, he can review it and make adjustments as needed.

A screenshot of a computer

Description automatically generated

### 5. Finishing a Task

* **Description:** Allows users to finish a task that is assigned to them.
* **Actor:** Lecturer
* **Preconditions:** The user must be logged in.
* **Basic Flow:**
  1. The user accesses the tasks section of the system.
  2. The user selects a task to view.
  3. The user answers the task and submits his answer.
* **Alternate Flows:**
  1. If the user does not accept the task changes, then no change is made to the database and the task is finished.



# Reference

## User Facing Functions

**1. Add Appendix**

* **Functionality:** Allows users to add new appendices to the system.
* **Example:** The user navigates to the 'Add Appendix' section, enters the details of the new appendix, and submits the form. The system adds the appendix to the database.

**2. Add Lecturer**

* **Functionality:** Allows administrators to add new lecturers to the system.
* **Example:** The administrator navigates to the 'Add Lecturer' section, enters the details of the new lecturer, and submits the form. The system adds the lecturer to the database and sends a confirmation email.

**3. Add Meta-Question**

* **Functionality:** Allows users to add new meta-questions to the system.
* **Example:** The user navigates to the 'Add Meta Question' section, enters the details of the new meta-question, and submits the form. The system adds the meta-question to the database.

**4. Add TA**

* **Functionality:** Allows administrators to add new TAs to the system.
* **Example:** The administrator navigates to the 'Add TA' section, enters the details of the new TA, and submits the form. The system adds the TA to the database and sends a confirmation email.

**5. Change Password**

* **Functionality:** Allows users to change their password.
* **Example:** The user navigates to the login section, enters their current password, and sets a new password. The system updates the user's password.

**6. Complete Task**

* **Functionality:** Marks a user-created task as complete.
* **Example:** The user views a created task, completes the required actions, and clicks Submit'. The system updates the task status to 'Completed'.

**7. Create Exam**

* **Functionality:** Allows users to create a new exam.
* **Example:** The user navigates to the 'Create Exam' section, enters the exam details, selects questions, and clicks 'Create Exam'. The system generates the exam and makes it available for students.

**8. Delete Appendix**

* **Functionality:** Allows users to delete an appendix from the system.
* **Example:** The user selects an appendix from the appendix list, clicks 'Delete Appendix', and confirms the action. The system removes the appendix from the database.

**9. Delete Question**

* **Functionality:** Allows users to delete a question from the system.
* **Example:** The user selects a question from the question list, clicks 'Delete Question', and confirms the action. The system removes the question from the database.

**10. Delete User**

* **Functionality:** Allows administrators to delete a user.
* **Example:** The administrator selects a user from the user list, clicks 'Delete User', and confirms the action. The system removes the user from the database.

**11. Edit Appendix**

* **Functionality:** Allows users to edit existing appendices.
* **Example:** The user selects an appendix from the list, navigates to 'Edit Appendix', updates the details, and clicks 'Save'. The system updates the appendix in the database.

**12. Edit Meta-Question**

* **Functionality:** Allows users to edit existing meta-questions.
* **Example:** The user selects a meta-question from the list, navigates to 'Edit Meta Question', updates the details, and clicks 'Save'. The system updates the meta-question in the database.

**13. Edit User**

* **Functionality:** Allows administrators to edit user details.
* **Example:** The administrator selects a user from the user list, navigates to 'Edit User', updates the user's details, and clicks 'Save'. The system updates the user's information in the database.

**14. Generate Task**

* **Functionality:** Generates a new task for the user.
* **Example:** The user navigates to the 'Tag Answers' section. The system creates a new ‘Tag Answers’ task for the user.

**15. Get All Appendices**

* **Functionality:** Retrieves a list of all appendices.
* **Example:** The user clicks on 'View All Appendices' in the questions management section. The system fetches and displays a list of all appendices.

**16. Get All Exams**

* **Functionality:** Retrieves a list of all exams.
* **Example:** The user clicks on 'View All Exams' in the exams management section. The system fetches and displays a list of all exams, including their status and assigned users.

**17. Get All Meta-Questions**

* **Functionality:** Retrieves a list of all meta-questions.
* **Example:** The user clicks on 'View All Meta Questions' in the questions management section. The system fetches and displays a list of all meta-questions.

**18. Get All Staff**

* **Functionality:** Retrieves a list of all staff members.
* **Example:** The user clicks on 'View All Staff' in the user management section. The system fetches and displays a list of all staff members.

**19. Get All Users**

* **Functionality:** Retrieves a list of all users.
* **Example:** The administrator clicks on 'View All Users' in the user management section. The system fetches and displays a list of all users.

**20. Get Meta-Questions for Appendix**

* **Functionality:** Retrieves meta-questions associated with a specific appendix.
* **Example:** The user selects an appendix and clicks on 'View Meta Questions'. The system displays all meta-questions associated with the appendix.

**21. Get Meta-Questions for Exam**

* **Functionality:** Retrieves meta-questions associated with a specific exam.
* **Example:** The user selects an exam and clicks on 'View Meta Questions'. The system displays all meta-questions associated with the exam.

**22. Remove Question from Exam**

* **Functionality:** Allows users to remove questions from an exam.
* **Example:** The user selects an exam, navigates to the question list, selects a question, and clicks 'Remove'. The system removes the question from the exam.

**23. Reset Password**

* **Functionality:** Allows users to reset their password.
* **Example:** The user navigates to the ‘user menu’ section and clicks 'Reset Password'.

**24. Sign In**

* **Functionality:** Allows users to sign in to the system.
* **Example:** The user navigates to the login page, enters their credentials, and clicks 'Sign In'. The system authenticates the user and grants access to the system.

**25. Sign Out**

* **Functionality:** Allows users to sign out of the system.
* **Example:** The user clicks on 'Sign Out'. The system logs the user out and redirects them to the login page.

**26. View Questions**

* **Functionality:** Allows users to view all questions in the system.
* **Example:** The user navigates to the 'View Questions' section. The system displays a list of all questions, organized by subject and keyword.

**27. View Tasks**

* **Functionality:** Allows users to view their tasks.
* **Example:** The user navigates to the dashboard. The system displays a list of tasks assigned to the user.

# Limitations

The current version of the Exam Management System has the following limitations:

* The system does not support offline functionality.
* Real-time collaboration on exam creation is not supported.
* Limited customization options for exam templates.

# Index

* **A:** Add Appendix, Add Lecturer, Add Meta-Question, Add TA
* **C:** Change Password, Complete Task, Create Exam
* **D:** Delete Appendix, Delete Question, Delete User
* **E:** Edit Appendix, Edit Meta-Question, Edit User
* **G:** Generate Task, Get All Appendices, Get All Exams, Get All Meta-Questions, Get All Staff, Get All Users, Get Meta-Questions for Appendix, Get Meta-Questions for Exam
* **R:** Remove Question from Exam, Reset Password
* **S:** Sign In, Sign Out
* **V:** View Questions, View Tasks