Exam Documentation (3 - 12 - 2024)

Exams API Documentation

Admin Endpoints

BaseURL: `/dashboard/`

1. Question Category Endpoints

List and Create Question Categories

- URL: `/question-categories/`
- Method: `GET`, `POST`
- Permissions: `IsAuthenticated`, `IsAdminUser`
- Query Params:
  - `year` (optional): Filter by year ID.
- Request Body (POST):

```
{
  "title": "string",
  "year": "integer"
}
```

- Response (GET):

```
[
  {
    "id": "integer",
    "title": "string",
    "year": "integer"
  }
]
```

- Response (POST):

```
{
  "id": "integer",
  "title": "string",
  "year": "integer"
}
```

Retrieve, Update, Delete Question Category

- URL: `/question-categories/<int:pk>/`
- Method: `GET`, `PUT`, `PATCH`, `DELETE`
- Permissions: `IsAuthenticated`, `IsAdminUser`
- Response (GET):

```
{
  "id": "integer",
  "title": "string",
  "year": "integer"
}
```

- Request Body (PUT/PATCH):

```
{
  "title": "string",
  "year": "integer"
}
```

- Response (DELETE):

```
{
  "detail": "Question category deleted successfully."
}
```

2. Question Endpoints

List and Create Questions

- URL: `/questions/`
- Method: `GET`, `POST`
- Permissions: `IsAuthenticated`, `IsAdminUser`
- Query Params:
  - `category__year`: Filter by category year.
  - `category`: Filter by category ID.
  - `unit`: Filter by unit ID.
  - `lesson`: Filter by lesson ID.
  - `is_active`: Filter by active status.
  - `difficulty`: Filter by difficulty level.
  - `search`: Search by question text.
- Request Body (POST):

```
{
  "text": "string",
  "image": "file",
  "points": "integer",
  "difficulty": "string",
  "category": "integer",
  "lesson": "integer",
  "unit": "integer",
  "is_active": "boolean",
  "answers": [
    {
      "text": "string",
      "image": "file",
      "is_correct": "boolean"
    }
  ]
}
```

- Response (GET):

```
{
  "count": "integer",
  "results": [
    {
      "id": "integer",
      "text": "string",
      "image": "string",
      "points": "integer",
      "difficulty": "string",
      "category": "integer",
      "lesson": "integer",
      "unit": "integer",
      "is_active": "boolean",
      "answers": [
        {
          "id": "integer",
          "text": "string",
          "image": "string"
        }
      ]
    }
  ],
  "statistics": {
    "num_active_questions": "integer",
    "num_easy_questions": "integer",
    "num_medium_questions": "integer",
    "num_hard_questions": "integer"
  }
}
```

- Response (POST):

```
{
  "id": "integer",
  "text": "string",
  "image": "string",
  "points": "integer",
  "difficulty": "string",
  "category": "integer",
  "lesson": "integer",
  "unit": "integer",
  "is_active": "boolean",
  "answers": [
    {
      "id": "integer",
      "text": "string",
      "image": "string",
      "is_correct": "boolean"
    }
  ]
}
```

Retrieve, Update, Delete Question

- URL: `/questions/<int:pk>/`
- Method: `GET`, `PUT`, `PATCH`, `DELETE`
- Permissions: `IsAuthenticated`, `IsAdminUser`
- Response (GET):

```
{
  "id": "integer",
  "text": "string",
  "image": "string",
  "points": "integer",
  "difficulty": "string",
  "category": "integer",
  "lesson": "integer",
  "unit": "integer",
  "is_active": "boolean",
  "answers": [
    {
      "id": "integer",
      "text": "string",
      "image": "string",
      "is_correct": "boolean"
    }
  ]
}
```

- Request Body (PUT/PATCH):

```
{
  "text": "string",
  "image": "file",
  "points": "integer",
  "difficulty": "string",
  "category": "integer",
  "lesson": "integer",
  "unit": "integer",
  "is_active": "boolean",
  "answers": [
    {
      "text": "string",
      "image": "file",
      "is_correct": "boolean"
    }
  ]
}
```

- Response (DELETE):

```
{
  "detail": "Question deleted successfully."
}
```

3. Answer Endpoints

List Answers for a Question

- URL: `/questions/<int:pk>/answers/`
- Method: `GET`
- Permissions: `IsAuthenticated`, `IsAdminUser`
- Response:

```
[
  {
    "id": "integer",
    "text": "string",
    "image": "string"
  }
]
```

List and Create Answers

- URL: `/answers/`
- Method: `GET`, `POST`
- Permissions: `IsAuthenticated`, `IsAdminUser`
- Request Body (POST):

```
{
  "text": "string",
  "image": "file",
  "is_correct": "boolean",
  "question": "integer"
}
```

- Response (GET):

```
[
  {
    "id": "integer",
    "text": "string",
    "image": "string",
    "is_correct": "boolean",
    "question": "integer"
  }
]
```

- Response (POST):

```
{
  "id": "integer",
  "text": "string",
  "image": "string",
  "is_correct": "boolean",
  "question": "integer"
}
```

Retrieve, Update, Delete Answer

- URL: `/answers/<int:pk>/`
- Method: `GET`, `PUT`, `PATCH`, `DELETE`
- Permissions: `IsAuthenticated`, `IsAdminUser`
- Response (GET):

```
{
  "id": "integer",
  "text": "string",
  "image": "string",
  "is_correct": "boolean",
  "question": "integer"
}
```

- Request Body (PUT/PATCH):

```
{
  "text": "string",
  "image": "file",
  "is_correct": "boolean",
  "question": "integer"
}
```

- Response (DELETE):

```
{
  "detail": "Answer deleted successfully."
}
```

4. Question Count Endpoint

Get Question Counts

- URL: `/questions/count/`
- Method: `GET`
- Permissions: `IsAuthenticated`
- Query Params:
  - `is_active`: Filter by active status.
  - `category`: Filter by category ID.
  - `lesson`: Filter by lesson ID.
  - `unit`: Filter by unit ID.
  - `course`: Filter by course ID.
- Response:

```
{
  "count": "integer",
  "easy": "integer",
  "medium": "integer",
  "hard": "integer"
}
```

5. Exam Endpoints

List and Create Exams

- URL: `/exams/`
- Method: `GET`, `POST`
- Permissions: `IsAuthenticated`, `IsAdminUser`
- Query Params:
  - `related_to`: Filter by related to (UNIT/LESSON).

    - `is_active`: Filter by active status.
    - `unit`: Filter by unit ID.
    - `lesson`: Filter by lesson ID.
    - `type`: Filter by exam type.
    - `status`: Filter by exam status (soon, active, finished).
- Request Body (POST):

  ```json
  {
    "title": "string",
    "description": "string",
    "related_to": "string",
    "unit": "integer",
    "lesson": "integer",
    "number_of_questions": "integer",
    "time_limit": "integer",
    "score": "float",
    "passing_percent": "integer",
    "start": "datetime",
    "end": "datetime",
    "number_of_allowed_trials": "integer",
    "type": "string",
    "easy_questions_count": "integer",
    "medium_questions_count": "integer",
    "hard_questions_count": "integer",
    "show_answers_after_finish": "boolean",
    "order": "integer",
    "is_active": "boolean",
    "allow_show_results_at": "datetime"
  }
  ```

- Response (GET):

  ```json
  [
    {
      "id": "integer",
      "title": "string",
      "description": "string",
      "related_to": "string",
      "unit": "integer",
      "lesson": "integer",
      "number_of_questions": "integer",
      "time_limit": "integer",
      "score": "float",
      "passing_percent": "integer",
      "start": "datetime",
      "end": "datetime",
      "number_of_allowed_trials": "integer",
      "type": "string",
      "easy_questions_count": "integer",
      "medium_questions_count": "integer",
  ```

```
    "hard_questions_count": "integer",
    "show_answers_after_finish": "boolean",
    "order": "integer",
    "is_active": "boolean",
    "created": "datetime",
    "status": "string",
    "allow_show_results_at": "datetime"
  }
]
```

- Response (POST):

```
{
  "id": "integer",
  "title": "string",
  "description": "string",
  "related_to": "string",
  "unit": "integer",
  "lesson": "integer",
  "number_of_questions": "integer",
  "time_limit": "integer",
  "score": "float",
  "passing_percent": "integer",
  "start": "datetime",
  "end": "datetime",
  "number_of_allowed_trials": "integer",
  "type": "string",
  "easy_questions_count": "integer",
  "medium_questions_count": "integer",
  "hard_questions_count": "integer",
  "show_answers_after_finish": "boolean",
  "order": "integer",
  "is_active": "boolean",
  "created": "datetime",
  "status": "string",
  "allow_show_results_at": "datetime"
}
```

Retrieve, Update, Delete Exam

- URL: `/exams/<int:pk>/`
- Method: `GET`, `PUT`, `PATCH`, `DELETE`
- Permissions: `IsAuthenticated`, `IsAdminUser`
- Response (GET):

```
{
  "id": "integer",
  "title": "string",
  "description": "string",
  "related_to": "string",
  "unit": "integer",
  "lesson": "integer",
  "number_of_questions": "integer",
  "time_limit": "integer",
  "score": "float",
  "passing_percent": "integer",
  "start": "datetime",
  "end": "datetime",
  "number_of_allowed_trials": "integer",
  "type": "string",
  "easy_questions_count": "integer",
  "medium_questions_count": "integer",
  "hard_questions_count": "integer",
  "show_answers_after_finish": "boolean",
  "order": "integer",
  "is_active": "boolean",
  "created": "datetime",
  "status": "string",
  "allow_show_results_at": "datetime"
}
```

- Request Body (PUT/PATCH):

```
{
  "title": "string",
  "description": "string",
  "related_to": "string",
  "unit": "integer",
  "lesson": "integer",
  "number_of_questions": "integer",
  "time_limit": "integer",
  "score": "float",
  "passing_percent": "integer",
  "start": "datetime",
  "end": "datetime",
  "number_of_allowed_trials": "integer",
  "type": "string",
  "easy_questions_count": "integer",
  "medium_questions_count": "integer",
  "hard_questions_count": "integer",
  "show_answers_after_finish": "boolean",
  "order": "integer",
  "is_active": "boolean",
  "allow_show_results_at": "datetime"
}
```

- Response (DELETE):

```
{
  "detail": "Exam deleted successfully."
}
```

6. Exam Questions Endpoints

Get Exam Questions

- URL: `/exams/<int:exam_id>/questions/`
- Method: `GET`
- Permissions: `IsAuthenticated`, `IsAdminUser`
- Response:

```
{
  "exam_id": "integer",
  "exam_title": "string",
  "questions": [
    {
      "id": "integer",
      "text": "string",
      "image": "string",
      "points": "integer",
      "difficulty": "string",
      "category": "integer",
      "lesson": "integer",
      "unit": "integer",
      "is_active": "boolean",
      "answers": [
        {
          "id": "integer",
          "text": "string",
          "image": "string"
        }
      ]
    }
  ]
}
```

Remove Exam Question

- URL: `/exams/<int:exam_id>/questions/<int:question_id>/`
- Method: `DELETE`
- Permissions: `IsAuthenticated`, `IsAdminUser`
- Response:

```
{
  "success": "Question removed from exam"
}
```

Add Exam Questions

- **URL:** `/exams/<int:exam_id>/add-questions/`
- **Method:** `POST`
- **Permissions:** `IsAuthenticated`, `IsAdminUser`
- **Request Body:**

```
{
  "questions_ids": [
    "integer"
  ]
}
```

- **Response:**

```
{
  "message": "Questions successfully added to the exam"
}
```

7. Random Exam Bank Endpoints

Manage Random Exam Bank

- **URL:** `/exams/<int:exam_id>/random-bank/`
- **Method:** `POST`, `GET`
- **Permissions:** `IsAuthenticated`, `IsAdminUser`
- **Request Body (POST):**

```
{
  "question_ids": [
    "integer"
  ]
}
```

- **Response (POST):**

```
{
  "message": "Questions successfully added to the random exam bank"
}
```

- **Response (GET):**

```
{
  "exam": "integer",
  "questions": [
    {
      "id": "integer",
      "text": "string",
      "image": "string",
      "points": "integer",
      "difficulty": "string",
      "category": "integer",
      "lesson": "integer",
      "unit": "integer",
      "is_active": "boolean",
      "answers": [
        {
          "id": "integer",
          "text": "string",
          "image": "string"
        }
      ]
    }
  ]
}
```

8. Exam Model Endpoints

List and Create Exam Models

- **URL:** `/exammodels/`
- **Method:** `GET`, `POST`
- **Permissions:** `IsAuthenticated`, `IsAdminUser`
- **Query Params:**
  - `is_active`: Filter by active status.
  - `exam`: Filter by exam ID.
- **Request Body (POST):**

```
{
  "exam": "integer",
  "title": "string",
  "is_active": "boolean"
}
```

- **Response (GET):**

```
[
  {
    "id": "integer",
    "exam": "integer",
    "title": "string",
    "is_active": "boolean",
    "created": "datetime"
  }
]
```

- Response (POST):

```
{
  "id": "integer",
  "exam": "integer",
  "title": "string",
  "is_active": "boolean",
  "created": "datetime"
}
```

Retrieve, Update, Delete Exam Model

- URL: `/exammodels/<int:pk>/`
- Method: `GET`, `PUT`, `PATCH`, `DELETE`
- Permissions: `IsAuthenticated`, `IsAdminUser`
- Response (GET):

```
{
  "id": "integer",
  "exam": "integer",
  "title": "string",
  "is_active": "boolean",
  "created": "datetime"
}
```

- Request Body (PUT/PATCH):

```
{
  "exam": "integer",
  "title": "string",
  "is_active": "boolean"
}
```

- Response (DELETE):

```
{
  "detail": "Exam model deleted successfully."
}
```

9. Exam Model Questions Endpoints

Get Exam Model Questions

- **URL:** `/exams/models/<int:exam_model_id>/questions/`
- **Method:** `GET`
- **Permissions:** `IsAuthenticated`, `IsAdminUser`
- **Response:**

  ```json
  {
    "exam_model_id": "integer",
    "exam_model_title": "string",
    "questions": [
      {
        "id": "integer",
        "text": "string",
        "difficulty": "string",
        "options": [
          {
            "id": "integer",
            "text": "string"
          }
        ]
      }
    ]
  }
  ```

Remove Question from Exam Model

- **URL:** `/exams/models/<int:exam_model_id>/questions/<int:question_id>/`
- **Method:** `DELETE`
- **Permissions:** `IsAuthenticated`, `IsAdminUser`
- **Response:**

  ```json
  {
    "message": "Question successfully removed from the exam model"
  }
  ```

Add Single Question to Exam Model

- **URL:** `/exams/<int:exam_id>/exam-model/<int:exam_model_id>/add-single-question/`
- **Method:** `POST`
- **Permissions:** `IsAuthenticated`, `IsAdminUser`
- **Request Body:**

  ```json
  {
    "question_id": "integer"
  }
  ```

- Response:

```
{
  "message": "Question added to the exam model successfully.",
  "model_id": "integer",
  "question_id": "integer",
  "question_text": "string",
  "current_questions_count": "integer",
  "max_questions_count": "integer"
}
```

10. Suggest Questions for Exam Model

Suggest Questions for Model

- URL: `/exams/<int:exam_id>/suggest-questions/`
- Method: `GET`
- Permissions: `IsAuthenticated`, `IsAdminUser`
- Response:

```
{
  "exam_id": "integer",
  "exam_title": "string",
  "questions": [
    {
      "id": "integer",
      "text": "string",
      "image": "string",
      "points": "integer",
      "difficulty": "string",
      "category": "integer",
      "lesson": "integer",
      "unit": "integer",
      "is_active": "boolean",
      "answers": [
        {
          "id": "integer",
          "text": "string",
          "image": "string"
        }
      ]
    }
  ]
}
```

11. Add Questions to Exam Model

Add Questions to Model

- **URL:** `/exams/<int:exam_id>/exam-model/<int:exam_model_id>/add-questions/`
- **Method:** `POST`
- **Permissions:** `IsAuthenticated`, `IsAdminUser`
- **Request Body:**

  ```
  {
    "question_ids": [
      "integer"
    ]
  }
  ```

- **Response:**

  ```
  {
    "message": "Questions added to the exam model successfully",
    "model_id": "integer",
    "title": "string",
    "questions_count": "integer"
  }
  ```

12. Results Endpoints

List Results

- **URL:** `/exams/results/`
- **Method:** `GET`
- **Permissions:** `IsAuthenticated`, `IsAdminUser`
- **Query Params:**
  - `student`: Filter by student ID.
  - `exam`: Filter by exam ID.
  - `is_succeeded`: Filter by success status.
  - `search`: Search by student name or exam title.
- **Response:**

  ```
  [
    {
      "id": "integer",
      "student": {
        "id": "integer",
        "name": "string"
      },
      "exam": {
        "id": "integer",
        "title": "string"
      },
      "score": "float",
  ```

```
    "exam_score": "float",
    "trial": "integer",
    "is_trials_finished": "boolean",
    "is_succeeded": "boolean",
    "added": "datetime"
  }
]
```

Get Student Result

- URL: `/exams/<int:exam_id>/students/<int:student_id>/result/`
- Method: `GET`
- Permissions: `IsAuthenticated`, `IsAdminUser`
- Response:

```
{
  "exam_id": "integer",
  "exam_title": "string",
  "exam_description": "string",
  "exam_score": "float",
  "student_id": "integer",
  "student_name": "string",
  "student_score": "float",
  "is_succeeded": "boolean",
  "student_trials": "integer",
  "is_trials_finished": "boolean",
  "student_answers": [
    {
      "question_id": "integer",
      "question_text": "string",
      "selected_answer": "string",
      "is_correct": "boolean"
    }
  ],
  "correct_answers": [
    {
      "question_id": "integer",
      "question_text": "string",
      "correct_answers": [
        "string"
      ]
    }
  ]
}
```

Students Endpoints

BaseURL: `/exams/`

1. Start Exam

Start an Exam

- URL: `/<int:exam_id>/start/`
- Method: `GET`
- Permissions: `IsAuthenticated`
- Response:

```json
{
  "exam_id": "integer",
  "exam_title": "string",
  "questions": [
    {
      "id": "integer",
      "text": "string",
      "image": "string",
      "points": "integer",
      "difficulty": "string",
      "category": "integer",
      "lesson": "integer",
      "unit": "integer",
      "is_active": "boolean",
      "answers": [
        {
          "id": "integer",
          "text": "string",
          "image": "string"
        }
      ],
      "correct_answers": [
        "string"
      ]
    }
  ],
  "exam_model": {
    "id": "integer",
    "title": "string"
  }
}
```

2. Submit Exam

Submit an Exam

- URL: `/<int:exam_id>/submit/`
- Method: `POST`

- **Permissions:** `IsAuthenticated`
- **Request Body:**

```
{
  "answers": [
    {
      "question_id": "integer",
      "selected_answer_id": "integer"
    }
  ],
  "exam_model_id": "integer"  // For RANDOM exams
}
```

- **Response:**

```
{
  "message": "Submission successful",
  "result": {
    "exam_id": "integer",
    "exam_title": "string",
    "exam_description": "string",
    "exam_score": "float",
    "student_id": "integer",
    "student_name": "string",
    "student_score": "float",
    "is_succeeded": "boolean",
    "student_trials": "integer",
    "is_trials_finished": "boolean",
    "student_answers": [
      {
        "question_id": "integer",
        "question_text": "string",
        "selected_answer": "string",
        "is_correct": "boolean"
      }
    ],
    "correct_answers": [
      {
        "question_id": "integer",
        "question_text": "string",
        "correct_answers": [
          "string"
        ]
      }
    ]
  }
}
```

3. Get My Exam Result

Get My Exam Result

- **URL:** `/<int:exam_id>/result/`
- **Method:** `GET`
- **Permissions:** `IsAuthenticated`
- **Response:**

```
{
  "exam_id": "integer",
  "exam_title": "string",
  "exam_description": "string",
  "exam_score": "float",
  "student_id": "integer",
  "student_name": "string",
  "student_score": "float",
  "is_succeeded": "boolean",
  "student_trials": "integer",
  "is_trials_finished": "boolean",
  "student_answers": [
    {
      "question_id": "integer",
      "question_text": "string",
      "selected_answer": "string",
      "is_correct": "boolean"
    }
  ],
  "correct_answers": [
    {
      "question_id": "integer",
      "question_text": "string",
      "correct_answers": [
        "string"
      ]
    }
  ]
}
```