

# Get application logs - GET /api/v1/admin/logs

**Mô tả:** Get recent application logs (admin only)

## Response

| Status Code | Description | Response Body (JSON Schema) | Example |

---|---|---|---

| 200 | Application logs | json { "message": "string" } | ``json { "message": "Success" }

```
| 401 | Unauthorized | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

```
| 403 | Forbidden | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

```
| 500 | Internal server error | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

## Seed database - POST /api/v1/admin/seed

**Mô tả:** Seed the database with initial users, questions, and exams (admin only)

### Response

| Status Code | Description | Response Body (JSON Schema) | Example |

---|---|---|---

```
| 200 | Database seeded successfully | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

```
| 401 | Unauthorized | ``json
{
  "message": "string"
}
```

```

    message : string
  }
  ... | ...json
{
  "message": "Success"
}
... |

| 403 | Forbidden | ...json
{
  "message": "string"
}
... | ...json
{
  "message": "Success"
}
... |

| 500 | Internal server error | ...json
{
  "message": "string"
}
... | ...json
{
  "message": "Success"
}
... |

```

## Login user - POST /api/v1/auth/login

**Mô tả:** Authenticate user and return JWT tokens

### Request Body

```

...json
{
  "email": "<email string>",
  "password": "<password string>"
}

```

#### JSON Schema:

```

{
  "properties": {
    "email": {
      "type": "string"
    },
    "password": {
      "minLength": 6,
      "type": "string"
    }
  },
  "required": [
    "email",
    "password"
  ],
  "type": "object"
}

```

#### Response

| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|

| 200 | Login successful | json { "message": "string" } | ``json{"message": "Success"}

```
| 400 | Bad request | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

```
| 401 | Invalid credentials | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

```
| 500 | Internal server error | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

## Logout user - POST /api/v1/auth/logout

\*\*Mô tả:\*\* Invalidate user's refresh token

### Response

| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|---|---|

```
| 200 | Logout successful | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

```
| 401 | Unauthorized | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

```
| 500 | Internal server error | ``json
```

```

{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

```

## Get current user profile - GET /api/v1/auth/profile

**Mô tả:** Get the profile of the currently authenticated user

### Response

Status Code	Description	Response Body (JSON Schema)	Example
-------------	-------------	-----------------------------	---------

Status Code	Description	Response Body (JSON Schema)	Example
-------------	-------------	-----------------------------	---------

200	User profile	```json	
-----	--------------	---------	--

```

{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

```

401	Unauthorized	```json	
-----	--------------	---------	--

```

{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

```

500	Internal server error	```json	
-----	-----------------------	---------	--

```

{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

```

## Refresh access token - POST /api/v1/auth/refresh

**Mô tả:** Get a new access token using refresh token

### Request Body

```

```json
{
  "refresh_token": "<refresh_token string>"
}

```

JSON Schema:

```
{
  "properties": {
    "refresh_token": {
      "type": "string"
    }
  },
  "required": [
    "refresh_token"
  ],
  "type": "object"
}
```

Response

Status Code	Description	Response Body (JSON Schema)	Example
200	Token refreshed successfully	<code>json { "message": "string" }</code>	<code>json { "message": "Success" }</code>

```

| 400 | Bad request | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 401 | Invalid refresh token | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 500 | Internal server error | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

```

## Register a new user - POST /api/v1/auth/register

\*\*Mô tả:\*\* Create a new user account

### Request Body

```

```json
{
  "email": "<email string>",
  "first_name": "<first_name string>",
  "last_name": "<last_name string>",
  "password": "<password string>",
  "username": "<username string>"
}

```

**JSON Schema:**

```

{
  "properties": {
    "email": {
      "type": "string"
    },
    "first_name": {
      "type": "string"
    },
    "last_name": {
      "type": "string"
    },
    "password": {
      "minLength": 6,
      "type": "string"
    },
    "username": {
      "maxLength": 50,
      "minLength": 3,
      "type": "string"
    }
  },
  "required": [
    "email",
    "first_name",
    "last_name",
    "password",
    "username"
  ],
  "type": "object"
}

```

## Response

| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|

| 201 | User created successfully | json { "message": "string" } | ``json { "message": "Success" }

| 400 | Bad request | ``json

```

{
  "message": "string"
}

```

`` | ``json

```

{
  "message": "Success"
}

```

`` |

| 409 | User already exists | ``json

```

{
  "message": "string"
}

```

`` | ``json

```

{
  "message": "Success"
}

```

`` |

| 500 | Internal server error | ``json

```

{
  "message": "string"
}

```

`` | ``json

{

```
    "message": "Success"
  }
  ... |
```

```
## Get exams list - GET /api/v1/exams
```

```
**Mô tả:** Get a paginated list of exams (admin sees all, users see assigned exams)
```

```
### Response
```

```
| Status Code | Description | Response Body (JSON Schema) | Example |
```

```
| ---|---|---|---|
```

```
| 200 | Exams list | ```json
```

```
{
  "properties": {
    "exams": {
      "items": {
        "$ref": "#/definitions/models.ExamResponse"
      },
      "type": "array"
    },
    "page": {
      "type": "integer"
    },
    "page_size": {
      "type": "integer"
    },
    "total": {
      "type": "integer"
    },
    "total_pages": {
      "type": "integer"
    }
  },
  "type": "object"
}
```

```
``` | ```json
```

```
{
  "exams": [
    {
      "created_at": "<created_at string>",
      "created_by": 0,
      "description": "<description string>",
      "duration": 0,
      "end_time": "<end_time string>",
      "id": 0,
      "is_active": true,
      "pass_score": 0,
      "questions": [
        {
          "content": "<content string>",
          "created_at": "<created_at string>",
          "created_by": 0,
          "difficulty": {},
          "explanation": "<explanation string>",
          "id": 0,
          "is_active": true,
          "options": [
            {
              "id": "<id string>",
              "text": "<text string>"
            }
          ]
        }
      ]
    }
  ]
}
```



```

    ],
    "points": 0,
    "tags": [
        "<tags_item string>"
    ],
    "time_limit": 0,
    "title": "<title string>",
    "type": {},
    "updated_at": "<updated_at string>"
  }
],
"start_time": "<start_time string>",
"status": {},
"title": "<title string>",
"total_points": 0,
"updated_at": "<updated_at string>",
"user_exam": {
  "attempt_count": 0,
  "completed_at": "<completed_at string>",
  "expires_at": "<expires_at string>",
  "id": 0,
  "max_attempts": 0,
  "started_at": "<started_at string>",
  "status": {},
  "time_left": 0
}
}
],
"page": 0,
"page_size": 0,
"total": 0,
"total_pages": 0
}
```

| 401 | Unauthorized | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
```

| 500 | Internal server error | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
```

## Create exam - POST /api/v1/exams

**Mô tả:** Create a new exam (admin only)

### Request Body
```json
{
  "description": "<description string>",
  "duration": 0
}

```

```
    duration : 0,
    "end_time": "<end_time string>",
    "pass_score": 0,
    "questions": [
      {
        "order": 0,
        "points": 0,
        "question_id": 0
      }
    ],
    "start_time": "<start_time string>",
    "title": "<title string>"
  }
}
```

#### JSON Schema:

```
{
  "properties": {
    "description": {
      "type": "string"
    },
    "duration": {
      "description": "in minutes",
      "minimum": 1,
      "type": "integer"
    },
    "end_time": {
      "type": "string"
    },
    "pass_score": {
      "maximum": 100,
      "minimum": 0,
      "type": "integer"
    },
    "questions": {
      "items": {
        "$ref": "#/definitions/services.ExamQuestionRequest"
      },
      "minItems": 1,
      "type": "array"
    },
    "start_time": {
      "type": "string"
    },
    "title": {
      "type": "string"
    }
  },
  "required": [
    "duration",
    "questions",
    "title"
  ],
  "type": "object"
}
```

## Response

| Status Code | Description | Response Body (JSON Schema) | Example |

| 201 | Exam created successfully | json { "message": "string" } | ``json { "message": "Success" }

| 400 | Bad request | ``json  
{

```

    "message": "string"
  }
  ... | ...json
  {
    "message": "Success"
  }
  ... |

| 401 | Unauthorized | ...json
{
  "message": "string"
}
... | ...json
{
  "message": "Success"
}
... |

| 403 | Forbidden | ...json
{
  "message": "string"
}
... | ...json
{
  "message": "Success"
}
... |

| 500 | Internal server error | ...json
{
  "message": "string"
}
... | ...json
{
  "message": "Success"
}
... |
```

## Delete exam - DELETE /api/v1/exams/{id}

**Mô tả:** Delete a specific exam (admin only)

### Response

Status Code	Description	Response Body (JSON Schema)	Example
-------------	-------------	-----------------------------	---------

---	---	---	---
-----	-----	-----	-----

200   Exam deleted successfully   ...json
{
"message": "string"
}
...   ...json
{
"message": "Success"
}
...

400   Bad request   ...json
{
"message": "string"
}
...   ...json
{

```

    "message": "Success"
  }
  ``` |

| 401 | Unauthorized | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 403 | Forbidden | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 404 | Exam not found | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 409 | Cannot delete exam with existing results | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 500 | Internal server error | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

```

## Get exam by ID - GET /api/v1/exams/{id}

**Mô tả:** Get a specific exam by its ID

### Response

Status Code	Description	Response Body (JSON Schema)	Example
-------------	-------------	-----------------------------	---------

---	---	---	---
-----	-----	-----	-----

| 200 | Exam details | ```json

```
{
  "message": "string"
}
```

``` | ```json

```
{
  "message": "Success"
}
```

``` |

| 400 | Bad request | ```json

```
{
  "message": "string"
}
```

``` | ```json

```
{
  "message": "Success"
}
```

``` |

| 401 | Unauthorized | ```json

```
{
  "message": "string"
}
```

``` | ```json

```
{
  "message": "Success"
}
```

``` |

| 403 | Exam not assigned to user | ```json

```
{
  "message": "string"
}
```

``` | ```json

```
{
  "message": "Success"
}
```

``` |

| 404 | Exam not found | ```json

```
{
  "message": "string"
}
```

``` | ```json

```
{
  "message": "Success"
}
```

``` |

| 500 | Internal server error | ```json

```
{
  "message": "string"
}
```

``` | ```json

```
{
  "message": "Success"
}
```

``` |

## Update exam - PUT /api/v1/exams/{id}

\*\*Mô tả:\*\* Update a specific exam (admin only)

```

### Request Body
```json
{
  "description": "<description string>",
  "duration": 0,
  "end_time": "<end_time string>",
  "pass_score": 0,
  "questions": [
    {
      "order": 0,
      "points": 0,
      "question_id": 0
    }
  ],
  "start_time": "<start_time string>",
  "status": {},
  "title": "<title string>"
}

```

#### JSON Schema:

```

{
  "properties": {
    "description": {
      "type": "string"
    },
    "duration": {
      "minimum": 1,
      "type": "integer"
    },
    "end_time": {
      "type": "string"
    },
    "pass_score": {
      "maximum": 100,
      "minimum": 0,
      "type": "integer"
    },
    "questions": {
      "items": {
        "$ref": "#/definitions/services.ExamQuestionRequest"
      },
      "minItems": 1,
      "type": "array"
    },
    "start_time": {
      "type": "string"
    },
    "status": {
      "$ref": "#/definitions/models.ExamStatus"
    },
    "title": {
      "type": "string"
    }
  },
  "required": [
    "duration",
    "questions",
    "status",
    "title"
  ],
  "type": "object"
}

```

Response

| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|---|

| 200 | Exam updated successfully | `json { "message": "string" }` | ```json { "message": "Success" }`

```
| 400 | Bad request | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

```
| 401 | Unauthorized | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

```
| 403 | Forbidden | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

```
| 404 | Exam not found | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

```
| 409 | Cannot update completed exam | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

```
| 500 | Internal server error | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
}
```

```

''' |

## Assign exam to users - POST /api/v1/exams/{id}/assign

**Mô tả:** Assign an exam to specific users (admin only)

### Request Body
```json
{
  "expires_at": "<expires_at string>",
  "max_attempts": 0,
  "user_ids": [
    0
  ]
}

```

#### JSON Schema:

```

{
  "properties": {
    "expires_at": {
      "type": "string"
    },
    "max_attempts": {
      "minimum": 1,
      "type": "integer"
    },
    "user_ids": {
      "items": {
        "type": "integer"
      },
      "minItems": 1,
      "type": "array"
    }
  },
  "required": [
    "user_ids"
  ],
  "type": "object"
}

```

## Response

| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|---|

| 200 | Exam assigned successfully | `json { "message": "string" }` | ````json {"message": "Success" }`

```

| 400 | Bad request | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 401 | Unauthorized | ```json
{
  "message": "string"
}

```



```
``` | ```json
{
  "message": "Success"
}
``` |
```

```
| 403 | Forbidden | ```json
{
  "message": "string"
}
```

```
``` | ```json
{
  "message": "Success"
}
``` |
```

```
| 404 | Exam not found | ```json
{
  "message": "string"
}
```

```
``` | ```json
{
  "message": "Success"
}
``` |
```

```
| 500 | Internal server error | ```json
{
  "message": "string"
}
```

```
``` | ```json
{
  "message": "Success"
}
``` |
```

## Start exam - POST /api/v1/exams/{id}/start

**Mô tả:** Start an assigned exam for the current user

### Response

Status Code	Description	Response Body (JSON Schema)	Example
-------------	-------------	-----------------------------	---------

---	---	---	---
-----	-----	-----	-----

200   Exam started successfully	```json	{	
		"properties": {	
		"questions": {	
		"items": {	
		"\$ref": "#/definitions/models.QuestionResponse"	
		},	
		"type": "array"	
		},	
		"time_left": {	
		"description": "in seconds",	
		"type": "integer"	
		},	
		"user_exam": {	
		"\$ref": "#/definitions/models.UserExamResponse"	
		}	
		},	
		"type": "object"	

```

}
``` | ```json
{
  "questions": [
    {
      "content": "<content string>",
      "created_at": "<created_at string>",
      "created_by": 0,
      "difficulty": {},
      "explanation": "<explanation string>",
      "id": 0,
      "is_active": true,
      "options": [
        {
          "id": "<id string>",
          "text": "<text string>"
        }
      ],
      "points": 0,
      "tags": [
        "<tags_item string>"
      ],
      "time_limit": 0,
      "title": "<title string>",
      "type": {},
      "updated_at": "<updated_at string>"
    }
  ],
  "time_left": 0,
  "user_exam": {
    "attempt_count": 0,
    "completed_at": "<completed_at string>",
    "expires_at": "<expires_at string>",
    "id": 0,
    "max_attempts": 0,
    "started_at": "<started_at string>",
    "status": {},
    "time_left": 0
  }
}
``` |

| 400 | Bad request | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 401 | Unauthorized | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 403 | Exam cannot be started | ```json
{
  "message": "string"
}

```

```

    | ``json
    {
      "message": "Success"
    }
    |

| 404 | Exam not assigned to user | ``json
{
  "message": "string"
}
| ``json
{
  "message": "Success"
}
|

| 500 | Internal server error | ``json
{
  "message": "string"
}
| ``json
{
  "message": "Success"
}
|

## Submit exam - POST /api/v1/exams/{id}/submit

**Mô tả:** Submit answers for an exam

### Request Body
``json
{
  "answers": [
    {
      "question_id": 0,
      "selected_options": [
        "<selected_options_item string>"
      ],
      "time_spent": 0
    }
  ]
}

```

#### JSON Schema:

```

{
  "properties": {
    "answers": {
      "items": {
        "$ref": "#/definitions/services.SubmitAnswerRequest"
      },
      "type": "array"
    }
  },
  "required": [
    "answers"
  ],
  "type": "object"
}

```

#### Response

| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|---|

| 200 | Exam submitted successfully | json { "message": "string" } | ``json { "message": "Success" }

| 400 | Bad request | ``json

```
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

| 401 | Unauthorized | ``json

```
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

| 403 | Exam cannot be submitted | ``json

```
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

| 404 | Exam not assigned to user | ``json

```
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

| 500 | Internal server error | ``json

```
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

## Get questions list - GET /api/v1/questions

\*\*Mô tả:\*\* Get a paginated list of questions with optional filtering

### Response

| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|---|

```
| 200 | Questions list | ```json
{
  "properties": {
    "page": {
      "type": "integer"
    },
    "page_size": {
      "type": "integer"
    },
    "questions": {
      "items": {
        "$ref": "#/definitions/models.QuestionResponse"
      },
      "type": "array"
    },
    "total": {
      "type": "integer"
    },
    "total_pages": {
      "type": "integer"
    }
  },
  "type": "object"
}
```

```
``` | ```json
{
  "page": 0,
  "page_size": 0,
  "questions": [
    {
      "content": "<content string>",
      "created_at": "<created_at string>",
      "created_by": 0,
      "difficulty": {},
      "explanation": "<explanation string>",
      "id": 0,
      "is_active": true,
      "options": [
        {
          "id": "<id string>",
          "text": "<text string>"
        }
      ],
      "points": 0,
      "tags": [
        "<tags_item string>"
      ],
      "time_limit": 0,
      "title": "<title string>",
      "type": {},
      "updated_at": "<updated_at string>"
    }
  ],
  "total": 0,
  "total_pages": 0
}
``` |
```

```
| 401 | Unauthorized | ```json
{
  "message": "string"
}
``` | ```json
{
```

```
"message": "Success"
}
``` |

| 500 | Internal server error | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |
```

## Create question - POST /api/v1/questions

\*\*Mô tả:\*\* Create a new question (admin only)

### Request Body

```
```json
{
  "content": "<content string>",
  "difficulty": {},
  "explanation": "<explanation string>",
  "options": [
    {
      "id": "<id string>",
      "is_correct": true,
      "text": "<text string>"
    }
  ],
  "points": 0,
  "tags": [
    "<tags_item string>"
  ],
  "time_limit": 0,
  "title": "<title string>",
  "type": {}
}
```

JSON Schema:

```

{
  "properties": {
    "content": {
      "type": "string"
    },
    "difficulty": {
      "$ref": "#/definitions/models.QuestionDifficulty"
    },
    "explanation": {
      "type": "string"
    },
    "options": {
      "items": {
        "$ref": "#/definitions/models.Option"
      },
      "minItems": 2,
      "type": "array"
    },
    "points": {
      "minimum": 1,
      "type": "integer"
    },
    "tags": {
      "items": {
        "type": "string"
      },
      "minItems": 1,
      "type": "array"
    },
    "time_limit": {
      "minimum": 10,
      "type": "integer"
    },
    "title": {
      "type": "string"
    },
    "type": {
      "$ref": "#/definitions/models.QuestionType"
    }
  },
  "required": [
    "content",
    "difficulty",
    "options",
    "tags",
    "title",
    "type"
  ],
  "type": "object"
}

```

## Response

| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|

| 201 | Question created successfully | json { "message": "string" } | ``json { "message": "Success" }

| 400 | Bad request | ``json

```

{
  "message": "string"
}

```

`` | ``json

```

{

```

```

    "message": "Success"
  }
  ... |

| 401 | Unauthorized | ```json
{
  "message": "string"
}
... | ```json
{
  "message": "Success"
}
... |

| 403 | Forbidden | ```json
{
  "message": "string"
}
... | ```json
{
  "message": "Success"
}
... |

| 500 | Internal server error | ```json
{
  "message": "string"
}
... | ```json
{
  "message": "Success"
}
... |

## Delete question - DELETE /api/v1/questions/{id}

**Mô tả:** Delete a specific question (admin only)

### Response
| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|---|---|

| 200 | Question deleted successfully | ```json
{
  "message": "string"
}
... | ```json
{
  "message": "Success"
}
... |

| 400 | Bad request | ```json
{
  "message": "string"
}
... | ```json
{
  "message": "Success"
}
... |

```



```
| 401 | Unauthorized | ```json
```

```
{  
  "message": "string"  
}  
``` | ```json  
{  
  "message": "Success"  
}  
``` |
```

```
| 403 | Forbidden | ```json
```

```
{  
  "message": "string"  
}  
``` | ```json  
{  
  "message": "Success"  
}  
``` |
```

```
| 404 | Question not found | ```json
```

```
{  
  "message": "string"  
}  
``` | ```json  
{  
  "message": "Success"  
}  
``` |
```

```
| 409 | Question is used in active exams | ```json
```

```
{  
  "message": "string"  
}  
``` | ```json  
{  
  "message": "Success"  
}  
``` |
```

```
| 500 | Internal server error | ```json
```

```
{  
  "message": "string"  
}  
``` | ```json  
{  
  "message": "Success"  
}  
``` |
```

```
## Get question by ID - GET /api/v1/questions/{id}
```

```
**Mô tả:** Get a specific question by its ID
```

```
### Response
```

```
| Status Code | Description | Response Body (JSON Schema) | Example |
```

```
|---|---|---|---|
```

```
| 200 | Question details | ```json
```

```
{  
  "message": "string"  
}  
``` | ```json
```

```

| 200 | Success | ``json
{
  "message": "Success"
}
```

| 400 | Bad request | ``json
{
  "message": "string"
}
```

| 200 | Success | ``json
{
  "message": "Success"
}
```

| 401 | Unauthorized | ``json
{
  "message": "string"
}
```

| 200 | Success | ``json
{
  "message": "Success"
}
```

| 404 | Question not found | ``json
{
  "message": "string"
}
```

| 200 | Success | ``json
{
  "message": "Success"
}
```

| 500 | Internal server error | ``json
{
  "message": "string"
}
```

```

## Update question - PUT /api/v1/questions/{id}

\*\*Mô tả:\*\* Update a specific question (admin only)

### Request Body

```

``json
{
  "content": "<content string>",
  "difficulty": {},
  "explanation": "<explanation string>",
  "is_active": true,
  "options": [
    {
      "id": "<id string>",
      "is_correct": true,
      "text": "<text string>"
    }
  ],

```

```

"points": 0,
"tags": [
  "<tags_item string>"
],
"time_limit": 0,
"title": "<title string>",
"type": {}
}

```

#### JSON Schema:

```

{
  "properties": {
    "content": {
      "type": "string"
    },
    "difficulty": {
      "$ref": "#/definitions/models.QuestionDifficulty"
    },
    "explanation": {
      "type": "string"
    },
    "is_active": {
      "type": "boolean"
    },
    "options": {
      "items": {
        "$ref": "#/definitions/models.Option"
      },
      "minItems": 2,
      "type": "array"
    },
    "points": {
      "minimum": 1,
      "type": "integer"
    },
    "tags": {
      "items": {
        "type": "string"
      },
      "minItems": 1,
      "type": "array"
    },
    "time_limit": {
      "minimum": 10,
      "type": "integer"
    },
    "title": {
      "type": "string"
    },
    "type": {
      "$ref": "#/definitions/models.QuestionType"
    }
  },
  "required": [
    "content",
    "difficulty",
    "options",
    "tags",
    "title",
    "type"
  ],
  "type": "object"
}

```

Response

| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|---|

| 200 | Question updated successfully | json { "message": "string" } | ``json { "message": "Success" }

```
| 400 | Bad request | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

```
| 401 | Unauthorized | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

```
| 403 | Forbidden | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

```
| 404 | Question not found | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

```
| 500 | Internal server error | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |
```

## Get random questions by tags - GET /api/v1/questions/random

\*\*Mô tả:\*\* Get random questions filtered by tags and difficulty level

### Response

| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|---|---|

| 200 | Random questions | ```json

```
{
  "message": "string"
}
```

``` | ```json

```
{
  "message": "Success"
}
```

``` |

| 400 | Bad request | ```json

```
{
  "message": "string"
}
```

``` | ```json

```
{
  "message": "Success"
}
```

``` |

| 401 | Unauthorized | ```json

```
{
  "message": "string"
}
```

``` | ```json

```
{
  "message": "Success"
}
```

``` |

| 500 | Internal server error | ```json

```
{
  "message": "string"
}
```

``` | ```json

```
{
  "message": "Success"
}
```

``` |

## Get question tags - GET /api/v1/questions/tags

**\*\*Mô tả:\*\*** Get all available question tags

### Response

| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|---|---|

| 200 | Tags list | ```json

```
{
  "message": "string"
}
```

``` | ```json

```
{
  "message": "Success"
}
```

``` |

| 401 | Unauthorized | ```json

```

{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 500 | Internal server error | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

## Get results list - GET /api/v1/results

**Mô tả:** Get a paginated list of exam results (admin sees all, users see their own)

### Response
| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|---|---|

| 200 | Results list | ```json
{
  "properties": {
    "page": {
      "type": "integer"
    },
    "page_size": {
      "type": "integer"
    },
    "results": {
      "items": {
        "$ref": "#/definitions/models.ResultResponse"
      },
      "type": "array"
    },
    "total": {
      "type": "integer"
    },
    "total_pages": {
      "type": "integer"
    }
  },
  "type": "object"
}
``` | ```json
{
  "page": 0,
  "page_size": 0,
  "results": [
    {
      "answers": [
        {
          "correct_options": [
            "<correct_options_item string>"
          ],
          "is correct": true,

```

```

    "points": 0,
    "question": {
      "content": "<content string>",
      "created_at": "<created_at string>",
      "created_by": 0,
      "difficulty": {},
      "explanation": "<explanation string>",
      "id": 0,
      "is_active": true,
      "options": [
        {
          "id": "<id string>",
          "text": "<text string>"
        }
      ],
      "points": 0,
      "tags": [
        "<tags_item string>"
      ],
      "time_limit": 0,
      "title": "<title string>",
      "type": {},
      "updated_at": "<updated_at string>"
    },
    "question_id": 0,
    "selected_options": [
      "<selected_options_item string>"
    ],
    "time_spent": 0
  }
],
"created_at": "<created_at string>",
"duration": 0,
"end_time": "<end_time string>",
"exam_id": 0,
"exam_title": "<exam_title string>",
"id": 0,
"max_points": 0,
"passed": true,
"start_time": "<start_time string>",
"total_points": 0,
"user": {
  "created_at": "<created_at string>",
  "email": "<email string>",
  "first_name": "<first_name string>",
  "id": 0,
  "is_active": true,
  "last_name": "<last_name string>",
  "role": {},
  "updated_at": "<updated_at string>",
  "username": "<username string>"
},
"attempt_count": 0,
"completed_at": "<completed_at string>",
"expires_at": "<expires_at string>",
"max_attempts": 0,
"started_at": "<started_at string>",
"status": {},
"time_left": 0,
"user_exam_id": 0,
"user_id": 0
}
],
"total": 0,
"total_pages": 0

```

```
}
``` |

| 401 | Unauthorized | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 500 | Internal server error | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |
```

## Get result by ID - GET /api/v1/results/{id}

\*\*Mô tả:\*\* Get a specific exam result by its ID

### Response

Status Code	Description	Response Body (JSON Schema)	Example
-------------	-------------	-----------------------------	---------

---	---	---	---
-----	-----	-----	-----

200   Result details	```json	{	
		"message": "string"	
		}	
	```	```json	
		{	
		"message": "Success"	
		}	
	```		

400   Bad request	```json	{	
		"message": "string"	
		}	
	```	```json	
		{	
		"message": "Success"	
		}	
	```		

401   Unauthorized	```json	{	
		"message": "string"	
		}	
	```	```json	
		{	
		"message": "Success"	
		}	
	```		

404   Result not found	```json		
------------------------	---------	--	--

/



```

    "message": "string"
  }
  ... | ...json
  {
    "message": "Success"
  }
  ... |

| 500 | Internal server error | ...json
{
  "message": "string"
}
... | ...json
{
  "message": "Success"
}
... |

## Get exam results - GET /api/v1/results/exam/{exam_id}

**Mô tả:** Get all results for a specific exam (admin only)

### Response
| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|---|---|

| 200 | Exam results | ...json
{
  "properties": {
    "page": {
      "type": "integer"
    },
    "page_size": {
      "type": "integer"
    },
    "results": {
      "items": {
        "$ref": "#/definitions/models.ResultResponse"
      },
      "type": "array"
    },
    "total": {
      "type": "integer"
    },
    "total_pages": {
      "type": "integer"
    }
  },
  "type": "object"
}
... | ...json
{
  "page": 0,
  "page_size": 0,
  "results": [
    {
      "answers": [
        {
          "correct_options": [
            "<correct_options_item string>"
          ],
          "is_correct": true,

```

```

    "points": 0,
    "question": {
      "content": "<content string>",
      "created_at": "<created_at string>",
      "created_by": 0,
      "difficulty": {},
      "explanation": "<explanation string>",
      "id": 0,
      "is_active": true,
      "options": [
        {
          "id": "<id string>",
          "text": "<text string>"
        }
      ],
      "points": 0,
      "tags": [
        "<tags_item string>"
      ],
      "time_limit": 0,
      "title": "<title string>",
      "type": {},
      "updated_at": "<updated_at string>"
    },
    "question_id": 0,
    "selected_options": [
      "<selected_options_item string>"
    ],
    "time_spent": 0
  }
],
"created_at": "<created_at string>",
"duration": 0,
"end_time": "<end_time string>",
"exam_id": 0,
"exam_title": "<exam_title string>",
"id": 0,
"max_points": 0,
"passed": true,
"start_time": "<start_time string>",
"total_points": 0,
"user": {
  "created_at": "<created_at string>",
  "email": "<email string>",
  "first_name": "<first_name string>",
  "id": 0,
  "is_active": true,
  "last_name": "<last_name string>",
  "role": {},
  "updated_at": "<updated_at string>",
  "username": "<username string>"
},
"attempt_count": 0,
"completed_at": "<completed_at string>",
"expires_at": "<expires_at string>",
"max_attempts": 0,
"started_at": "<started_at string>",
"status": {},
"time_left": 0,
"user_exam_id": 0,
"user_id": 0
}
],
"total": 0,
"total_pages": 0

```

```

}
``` |

| 400 | Bad request | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 401 | Unauthorized | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 403 | Forbidden | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 500 | Internal server error | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

```

## Get current user's results - GET /api/v1/results/my-results

**\*\*Mô tả:\*\*** Get exam results for the currently authenticated user

### Response

Status Code	Description	Response Body (JSON Schema)	Example
-------------	-------------	-----------------------------	---------

|--|--|--|--|

200   User's results   ```json			
{			
"properties": {			
"page": {			
"type": "integer"			
},			
"page_size": {			
"type": "integer"			
},			
"results": {			
"items": {			
"\$ref": "#/definitions/models.ResultResponse"			

```

    },
    "type": "array"
  },
  "total": {
    "type": "integer"
  },
  "total_pages": {
    "type": "integer"
  }
},
"type": "object"
}
... | ...json
{
  "page": 0,
  "page_size": 0,
  "results": [
    {
      "answers": [
        {
          "correct_options": [
            "<correct_options_item string>"
          ],
          "is_correct": true,
          "points": 0,
          "question": {
            "content": "<content string>",
            "created_at": "<created_at string>",
            "created_by": 0,
            "difficulty": {},
            "explanation": "<explanation string>",
            "id": 0,
            "is_active": true,
            "options": [
              {
                "id": "<id string>",
                "text": "<text string>"
              }
            ],
            "points": 0,
            "tags": [
              "<tags_item string>"
            ],
            "time_limit": 0,
            "title": "<title string>",
            "type": {},
            "updated_at": "<updated_at string>"
          },
          "question_id": 0,
          "selected_options": [
            "<selected_options_item string>"
          ],
          "time_spent": 0
        }
      ],
      "created_at": "<created_at string>",
      "duration": 0,
      "end_time": "<end_time string>",
      "exam_id": 0,
      "exam_title": "<exam_title string>",
      "id": 0,
      "max_points": 0,
      "passed": true,
      "start_time": "<start_time string>",
      "total_points": 0,

```

```

    "user": {
      "created_at": "<created_at string>",
      "email": "<email string>",
      "first_name": "<first_name string>",
      "id": 0,
      "is_active": true,
      "last_name": "<last_name string>",
      "role": {},
      "updated_at": "<updated_at string>",
      "username": "<username string>"
    },
    "attempt_count": 0,
    "completed_at": "<completed_at string>",
    "expires_at": "<expires_at string>",
    "max_attempts": 0,
    "started_at": "<started_at string>",
    "status": {},
    "time_left": 0,
    "user_exam_id": 0,
    "user_id": 0
  }
],
"total": 0,
"total_pages": 0
}

```

```

``` |

```

```

| 401 | Unauthorized | ```json

```

```

{
  "message": "string"
}

```

```

``` | ```json

```

```

{
  "message": "Success"
}

```

```

``` |

```

```

| 500 | Internal server error | ```json

```

```

{
  "message": "string"
}

```

```

``` | ```json

```

```

{
  "message": "Success"
}

```

```

``` |

```

```

## Get current user's statistics - GET /api/v1/results/my-statistics

```

```

**Mô tả:** Get exam statistics for the currently authenticated user

```

```

### Response

```

```

| Status Code | Description | Response Body (JSON Schema) | Example |

```

```

|---|---|---|---|

```

```

| 200 | User statistics | ```json

```

```

{
  "message": "string"
}

```

```

``` | ```json

```

```

{
  "message": "Success"
}

```

```

\

```

```

f
``` |

| 401 | Unauthorized | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 500 | Internal server error | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

## Get statistics - GET /api/v1/results/statistics

**Mô tả:** Get comprehensive exam, user, and question statistics (admin only)

### Response
| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|---|---|

| 200 | Statistics data | ```json
{
  "properties": {
    "exam_statistics": {
      "items": {
        "$ref": "#/definitions/models.ExamStatistics"
      },
      "type": "array"
    },
    "overall_stats": {
      "$ref": "#/definitions/services.OverallStatistics"
    },
    "question_statistics": {
      "items": {
        "$ref": "#/definitions/models.QuestionStatistics"
      },
      "type": "array"
    },
    "user_statistics": {
      "items": {
        "$ref": "#/definitions/models.UserStatistics"
      },
      "type": "array"
    }
  },
  "type": "object"
}
``` | ```json
{
  "exam_statistics": [
    {
      "average_duration": 0,

```

```

        "exam_id": 0,
        "exam_title": "<exam_title string>",
        "failed_attempts": 0,
        "passed_attempts": 0,
        "total_attempts": 0
    }
],
"overall_stats": {
    "average_duration": 0,
    "total_attempts": 0,
    "total_exams": 0,
    "total_time_spent": 0,
    "total_users": 0
},
"question_statistics": [
    {
        "average_time_spent": 0,
        "correct_attempts": 0,
        "question_id": 0,
        "question_title": "<question_title string>",
        "total_attempts": 0,
        "wrong_attempts": 0
    }
],
"user_statistics": [
    {
        "failed_exams": 0,
        "passed_exams": 0,
        "total_exams": 0,
        "total_time_spent": 0,
        "user_id": 0,
        "username": "<username string>"
    }
]
}
```

| 401 | Unauthorized | ```json
{
    "message": "string"
}
``` | ```json
{
    "message": "Success"
}
```

| 403 | Forbidden | ```json
{
    "message": "string"
}
``` | ```json
{
    "message": "Success"
}
```

| 500 | Internal server error | ```json
{
    "message": "string"
}
``` | ```json
{
    "message": "Success"
}
```

```

```

    |

## Get users list - GET /api/v1/users

**Mô tả:** Get a paginated list of all users (admin only)

### Response
| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|---|---|

| 200 | Users list | ```json
{
  "properties": {
    "page": {
      "type": "integer"
    },
    "page_size": {
      "type": "integer"
    },
    "total": {
      "type": "integer"
    },
    "total_pages": {
      "type": "integer"
    },
    "users": {
      "items": {
        "$ref": "#/definitions/models.UserResponse"
      },
      "type": "array"
    }
  },
  "type": "object"
}
``` | ```json
{
  "page": 0,
  "page_size": 0,
  "total": 0,
  "total_pages": 0,
  "users": [
    {
      "created_at": "<created_at string>",
      "email": "<email string>",
      "first_name": "<first_name string>",
      "id": 0,
      "is_active": true,
      "last_name": "<last_name string>",
      "role": {},
      "updated_at": "<updated_at string>",
      "username": "<username string>"
    }
  ]
}
``` |

| 401 | Unauthorized | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
```

```



```
}
``` |

| 403 | Forbidden | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |
```

```
| 500 | Internal server error | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |
```

## Delete user - DELETE /api/v1/users/{id}

**Mô tả:** Delete a specific user (admin only)

### Response

Status Code	Description	Response Body (JSON Schema)	Example
-------------	-------------	-----------------------------	---------

---	---	---	---
-----	-----	-----	-----

200   User deleted successfully	```json	{	
		"message": "string"	
		}	
```	```json	{	
		"message": "Success"	
		}	
```			

400   Bad request	```json	{	
		"message": "string"	
		}	
```	```json	{	
		"message": "Success"	
		}	
```			

401   Unauthorized	```json	{	
		"message": "string"	
		}	
```	```json	{	
		"message": "Success"	
		}	
```			

403   Forbidden	```json		
-----------------	---------	--	--

```
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 404 | User not found | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 500 | Internal server error | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |
```

## Get user by ID - GET /api/v1/users/{id}

\*\*Mô tả:\*\* Get a specific user by their ID (admin only)

### Response

Status Code	Description	Response Body (JSON Schema)	Example
-------------	-------------	-----------------------------	---------

---	---	---	---
-----	-----	-----	-----

200	User details	```json	
-----	--------------	---------	--

```
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |
```

400	Bad request	```json	
-----	-------------	---------	--

```
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |
```

401	Unauthorized	```json	
-----	--------------	---------	--

```
{
  "message": "string"
}
``` | ```json
```

,

```

{
  "message": "Success"
}
``` |

| 403 | Forbidden | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 404 | User not found | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

| 500 | Internal server error | ```json
{
  "message": "string"
}
``` | ```json
{
  "message": "Success"
}
``` |

```

## Update user - PUT /api/v1/users/{id}

**Mô tả:** Update a specific user (admin only)

### Request Body

```

```json
{
  "email": "<email string>",
  "first_name": "<first_name string>",
  "is_active": true,
  "last_name": "<last_name string>",
  "role": {},
  "username": "<username string>"
}

```

**JSON Schema:**

```

{
  "properties": {
    "email": {
      "type": "string"
    },
    "first_name": {
      "type": "string"
    },
    "is_active": {
      "type": "boolean"
    },
    "last_name": {
      "type": "string"
    },
    "role": {
      "$ref": "#/definitions/models.UserRole"
    },
    "username": {
      "maxLength": 50,
      "minLength": 3,
      "type": "string"
    }
  },
  "required": [
    "email",
    "first_name",
    "last_name",
    "role",
    "username"
  ],
  "type": "object"
}

```

## Response

| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|

| 200 | User updated successfully | json { "message": "string" } | ``json { "message": "Success" }

| 400 | Bad request | ``json

```

{
  "message": "string"
}

```

`` | ``json

```

{
  "message": "Success"
}

```

`` |

| 401 | Unauthorized | ``json

```

{
  "message": "string"
}

```

`` | ``json

```

{
  "message": "Success"
}

```

`` |

| 403 | Forbidden | ``json

```

{
  "message": "string"
}

```

```

` `` | `` `json
{
  "message": "Success"
}
` `` |

| 404 | User not found | `` `json
{
  "message": "string"
}
` `` | `` `json
{
  "message": "Success"
}
` `` |

| 409 | Email or username already taken | `` `json
{
  "message": "string"
}
` `` | `` `json
{
  "message": "Success"
}
` `` |

| 500 | Internal server error | `` `json
{
  "message": "string"
}
` `` | `` `json
{
  "message": "Success"
}
` `` |

```

## Change user password - POST /api/v1/users/change-password

**\*\*Mô tả:\*\*** Change the password of the currently authenticated user

### Request Body

```

` `` `json
{
  "current_password": "<current_password string>",
  "new_password": "<new_password string>"
}

```

**JSON Schema:**

```
{
  "properties": {
    "current_password": {
      "type": "string"
    },
    "new_password": {
      "minLength": 6,
      "type": "string"
    }
  },
  "required": [
    "current_password",
    "new_password"
  ],
  "type": "object"
}
```

Response

| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|---|---|

| 200 | Password changed successfully | json { "message": "string" } | ``json {"message": "Success"}

| 400 | Bad request | ``json

```
{
  "message": "string"
}
```

`` | ``json

```
{
  "message": "Success"
}
```

`` |

| 401 | Unauthorized or incorrect current password | ``json

```
{
  "message": "string"
}
```

`` | ``json

```
{
  "message": "Success"
}
```

`` |

| 500 | Internal server error | ``json

```
{
  "message": "string"
}
```

`` | ``json

```
{
  "message": "Success"
}
```

`` |

## Get user profile - GET /api/v1/users/profile

\*\*Mô tả:\*\* Get the profile of the currently authenticated user

### Response

| Status Code | Description | Response Body (JSON Schema) | Example |

|---|---|---|---|

```

| 200 | User profile | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |

| 401 | Unauthorized | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |

| 500 | Internal server error | ``json
{
  "message": "string"
}
`` | ``json
{
  "message": "Success"
}
`` |

```

## Update user profile - PUT /api/v1/users/profile

**Mô tả:** Update the profile of the currently authenticated user

### Request Body

```

``json
{
  "first_name": "<first_name string>",
  "last_name": "<last_name string>",
  "username": "<username string>"
}

```

**JSON Schema:**

```
{
  "properties": {
    "first_name": {
      "type": "string"
    },
    "last_name": {
      "type": "string"
    },
    "username": {
      "maxLength": 50,
      "minLength": 3,
      "type": "string"
    }
  },
  "required": [
    "first_name",
    "last_name",
    "username"
  ],
  "type": "object"
}
```

Response

Status Code	Description	Response Body (JSON Schema)	Example
200	Profile updated successfully	<code>json { "message": "string" }</code>	<code>``json {"message": "Success" }</code>



| 400 | Bad request | ```json

```
{  
  "message": "string"  
}
```

``` | ```json

```
{  
  "message": "Success"  
}
```

``` |

| 401 | Unauthorized | ```json

```
{  
  "message": "string"  
}
```

``` | ```json

```
{  
  "message": "Success"  
}
```

``` |

| 409 | Username already taken | ```json

```
{  
  "message": "string"  
}
```

``` | ```json

```
{  
  "message": "Success"  
}
```

``` |

| 500 | Internal server error | ```json

```
{  
  "message": "string"  
}
```

``` | ```json

```
{  
  "message": "Success"  
}
```

``` |