ACTIVITY 6 - WS

JOAQUIM PICÓ MORA, IAN PALACÍN ALIANA, SERGI SIMÓN BALCELLS

Created: 2021-01-11 Mon 01:33

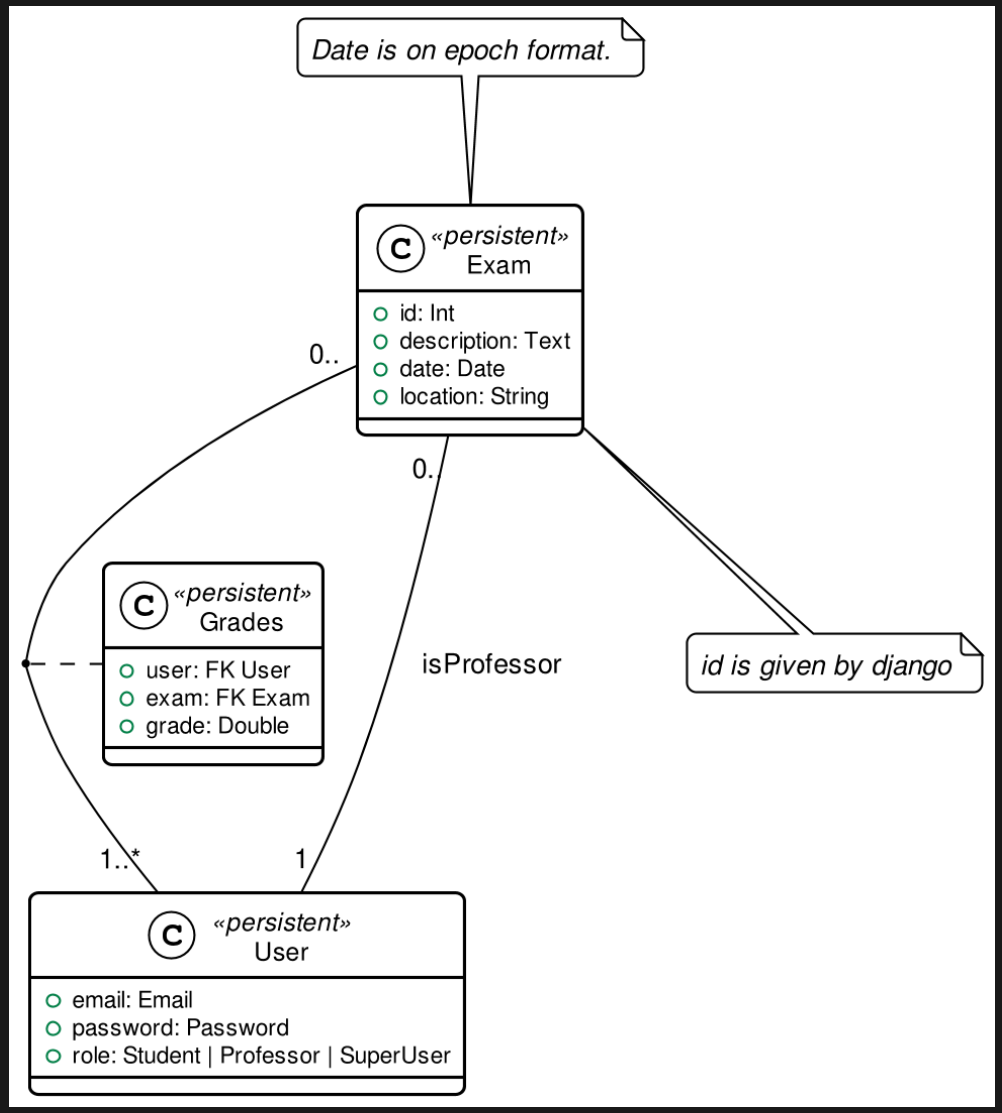
1 INTRODUCTION

In this document is it specified all the endpoints and which responses do they return, as well as a defense of which technologies we have used.

Additionally, it can be found a table containing the REST API developed, as well as a class diagram of the database model used by the API.

The integration can be found at RMI Project, at the branch `integration`. On the other hand, the Web Service can be found at WS Project.

2 UML


Date is on epoch format.
«persistent»
Exam
id: Int
description: Text
date: Date
location: String
0..
0..
«persistent»
Grades
user: FK User
exam: FK Exam
grade: Double
isProfessor
id is given by django
1..*
1
«persistent»
User
email: Email
password: Password
role: Student | Professor | SuperUser

2.1 EXAM

Exam is the class that holds all the Exam information. It stores the a description, a date and a location of an exam.

2.2 USER

User is the class that stores the information of a user that is making use of our system. It’s the default implementation of the Django User class.

2.3 GRADES

This class it’s the one that stores grades of exams made by users. It holds two foreign keys to the exam that belongs the grade, as well as the student.

3 ENDPOINT TABLE

Method	URL	What	Status code
get	exam/	List all the exams	200
get	exam/{exam}/	Detail of an exam	200, 404
get	exam/search?description={text}/	Search a description	200

post	exam/	Creates an exam.	201, 403, 401
put	exam/{exam}/	Modify all fields of an exam	200, 403, 401
patch	exam/{exam}/	Partial update.	200, 403, 401

delete	exam/{exam}/	Deletes if user is professor and has no grades	204, 403, 401
post	grades/	Uploads an exam.	201, 403, 401
get	grades/{user}/user/	List all grades of a user	200

get	grades/	List all grades.	200
get	grades/{grade-id}	Detail a grade.	200, 404
put	grades/{grade-id}	Updates a grade.	200, 403, 401

patch	grades/{grade-id}	Partially updates a grade.	200, 403, 401
delete	grades/{grade-id}	Deletes a grade.	204, 403, 401

post	auth/login/	Logins	201, 403, 401
get	auth/logout/	Logouts	200
post	auth/logout/	Logout	201, 403, 401

post	auth/password/change/	Password change.
post	auth/password/reset/	Password reset by email confirmation. Needs Email configuration
post	auth/password/reset/confirm/	Password Confirmation

post	auth/registration/	Register a new user.	201, 403, 401
post	auth/registration/verify-email	Verifies email. Needs Email configuration	201, 403, 401
get	auth/user/	Reads User. Needs authentication	200

put	auth/user/	Updates User	200, 403, 401
patch	auth/user/	Partial update.	200, 403, 401
get	user/{user}/	Gets user with pk.	200, 404

4 SCREENSHOTS

The screenshots are for the most important cases, there are endpoints that has been omitted, like user password change.

Note that due to a bug in the docs viewer, as deleting an object only returns a status code without any data, it does not correctly show that the status code is 204. Instead, only shows “undefined”, even though it is properly deleted from the database.

4.1 AUTHENTICATION

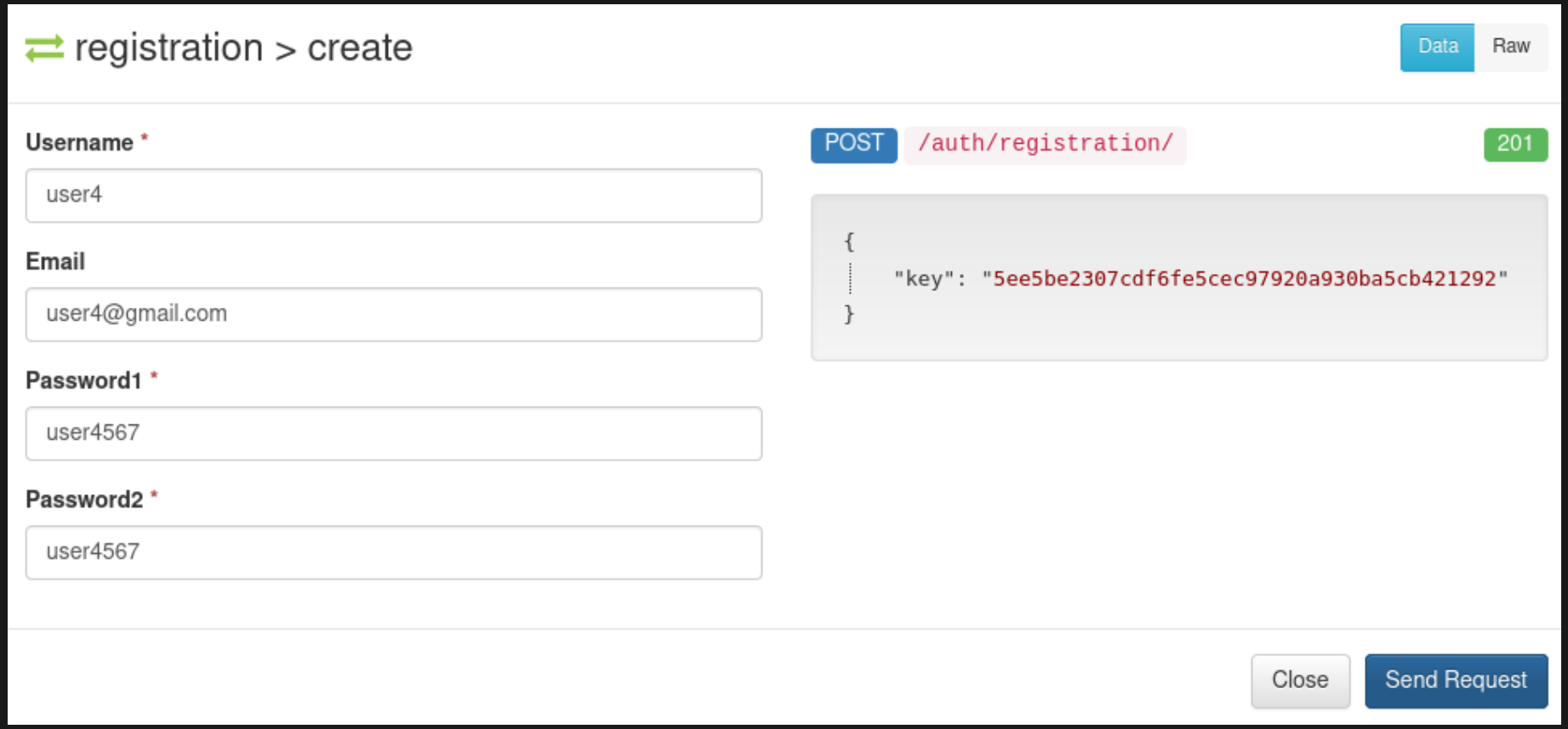


Figure 2: Register

Figure 3: Login

4.2 EXAM

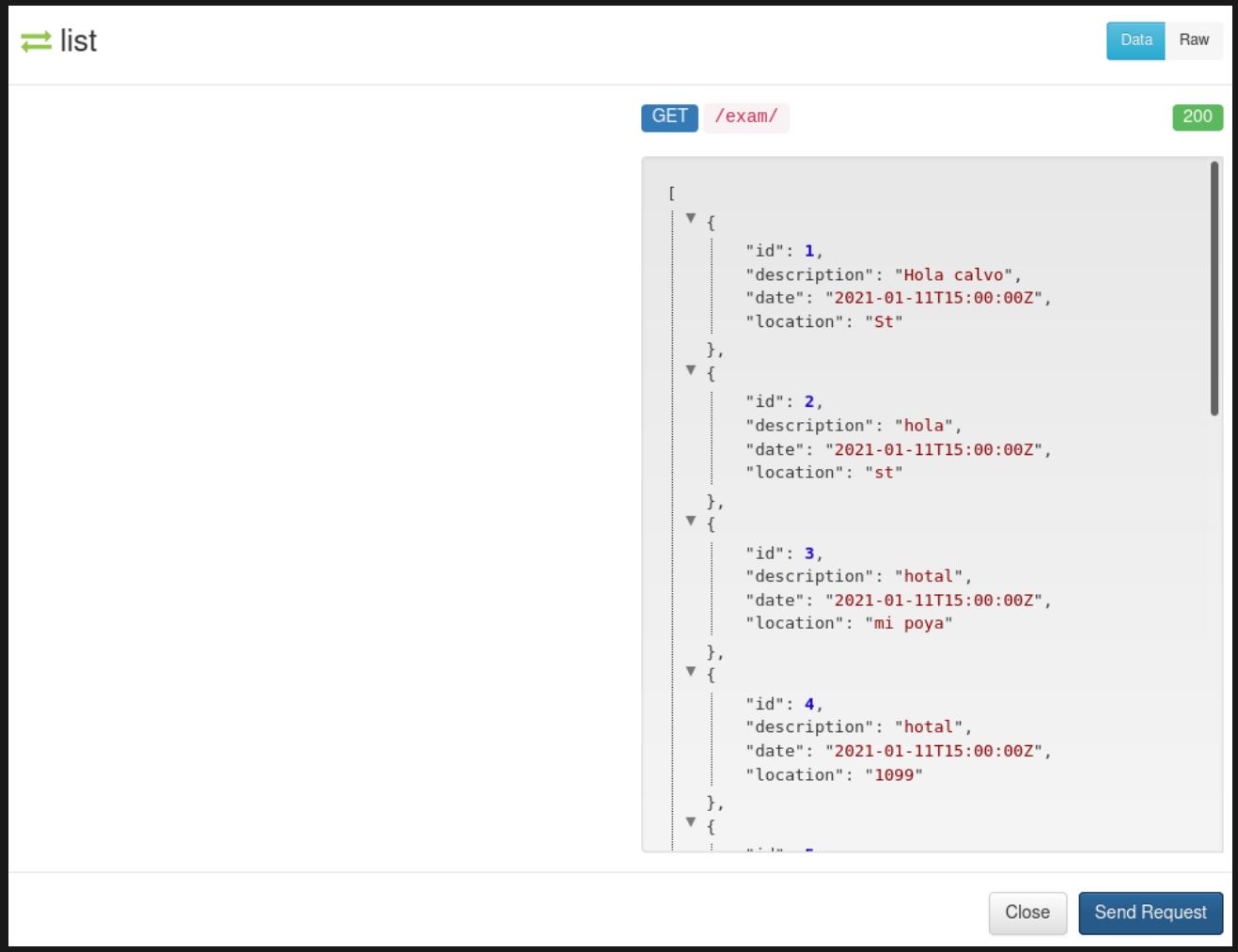


Figure 4: List exams

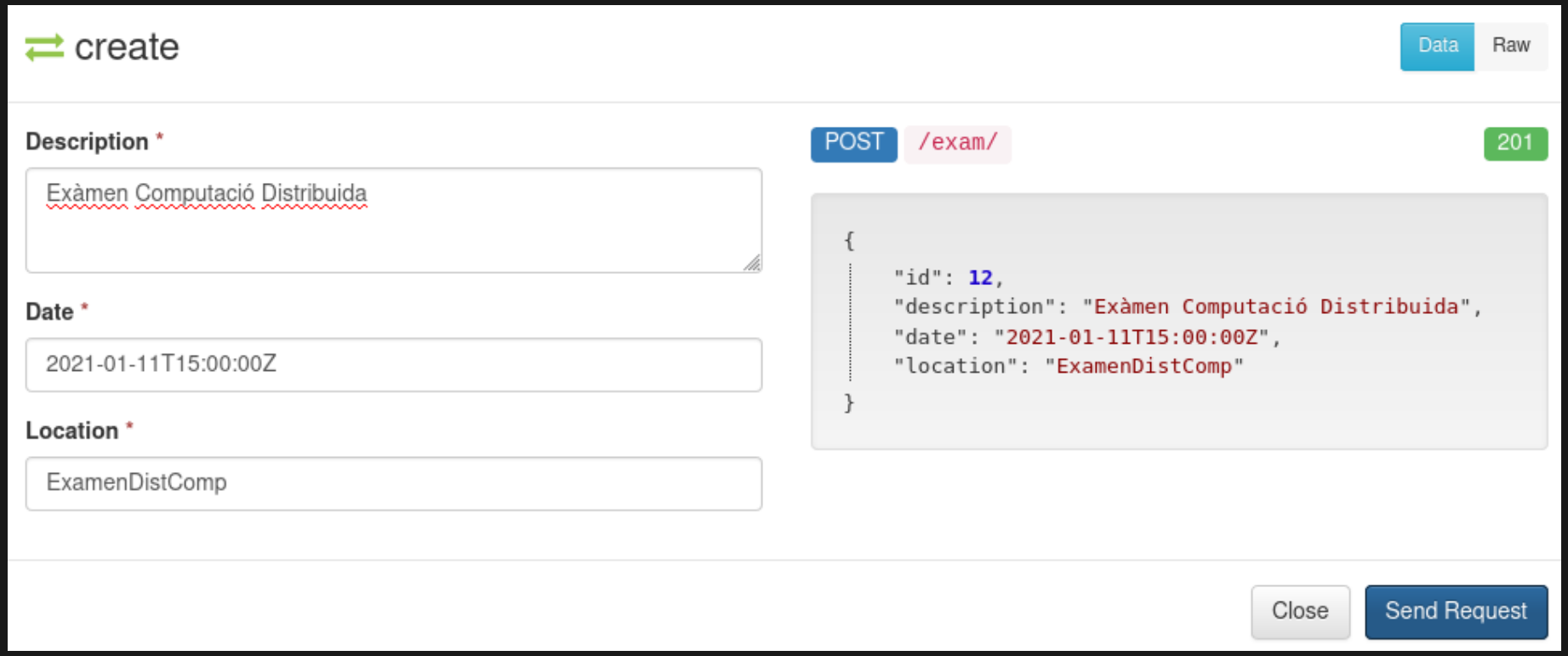


Figure 5: Create exam

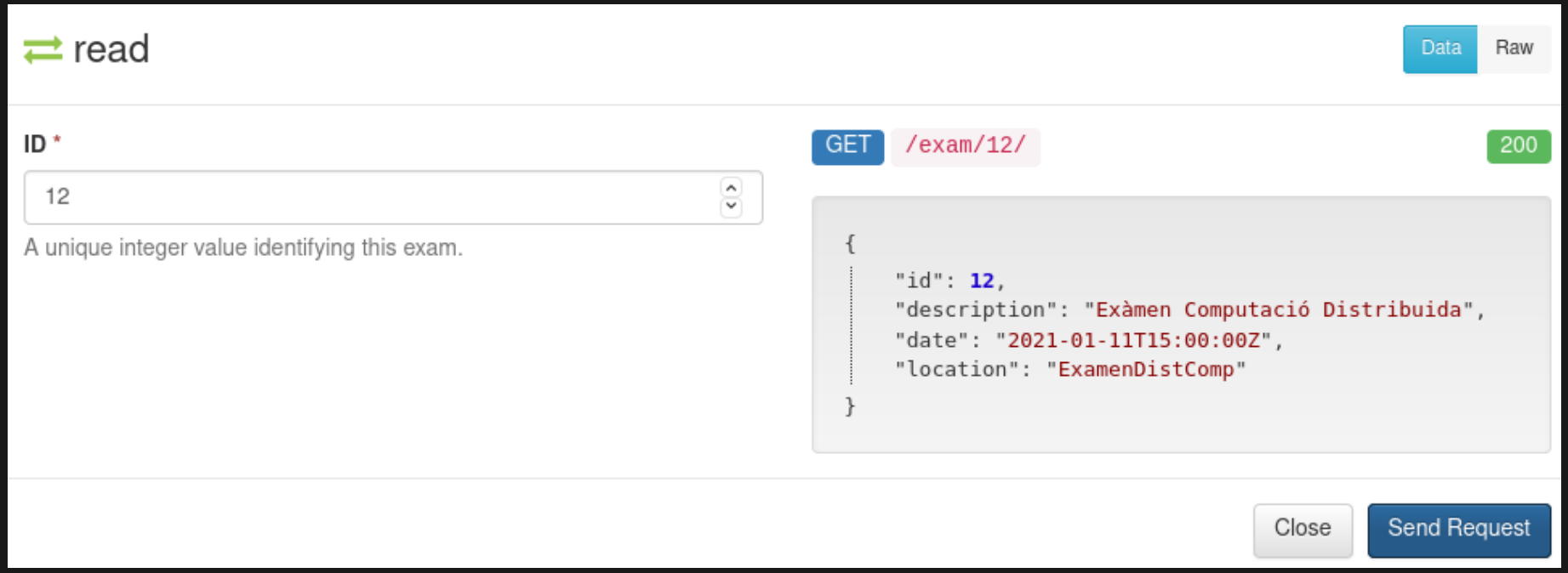


Figure 6: Read exam

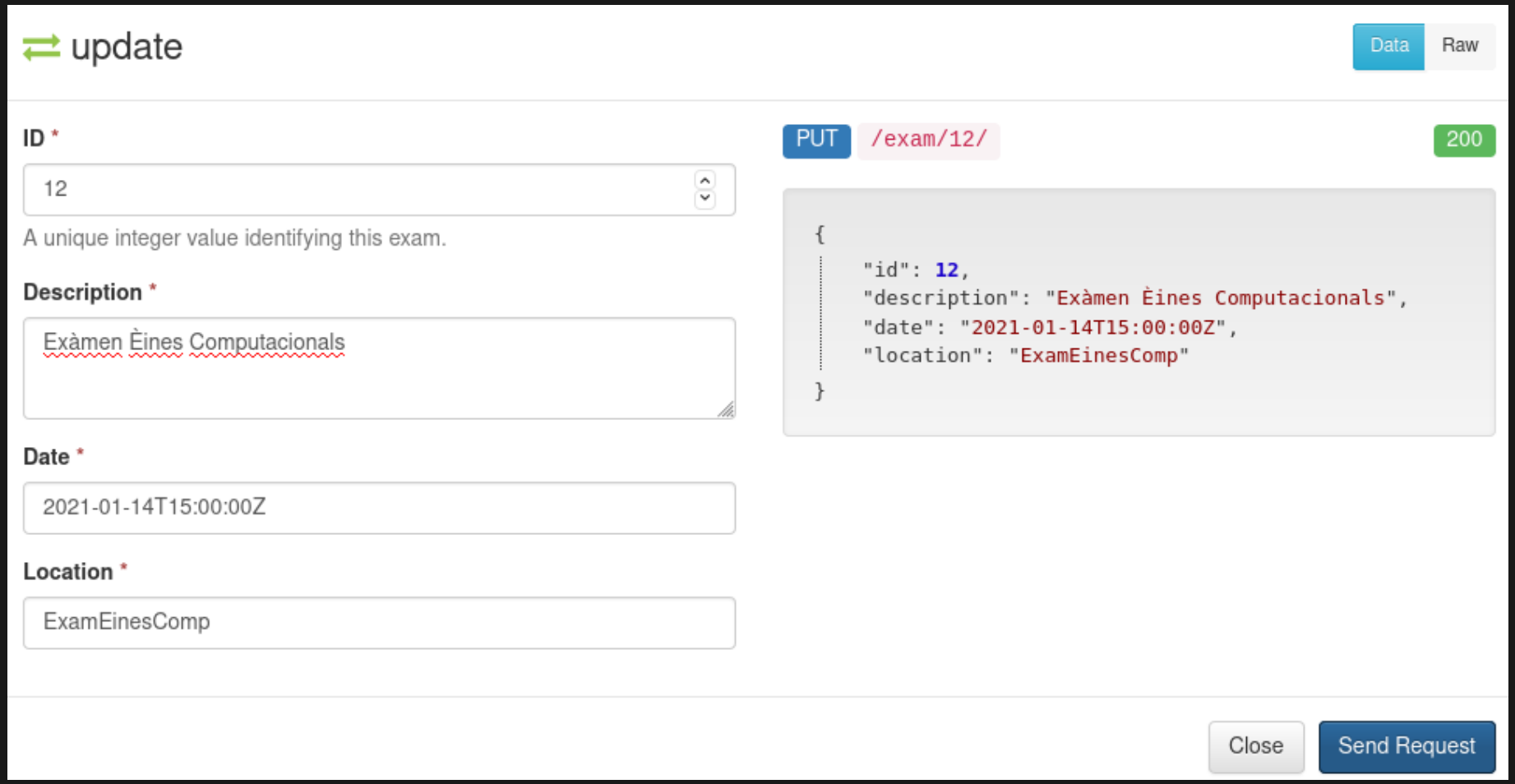


Figure 7: Update exam

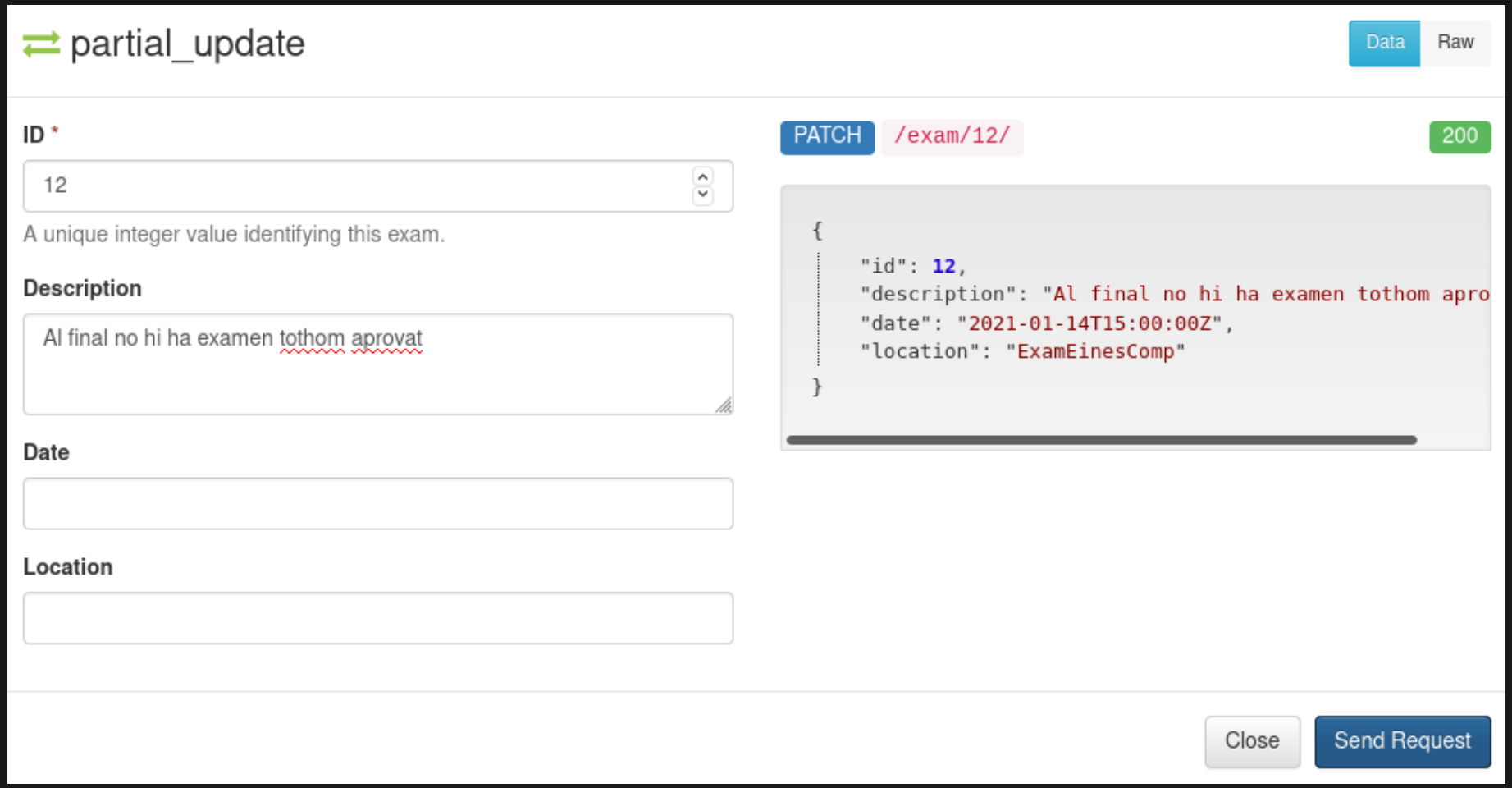


Figure 8: Patch exam

Figure 9: Delete exam

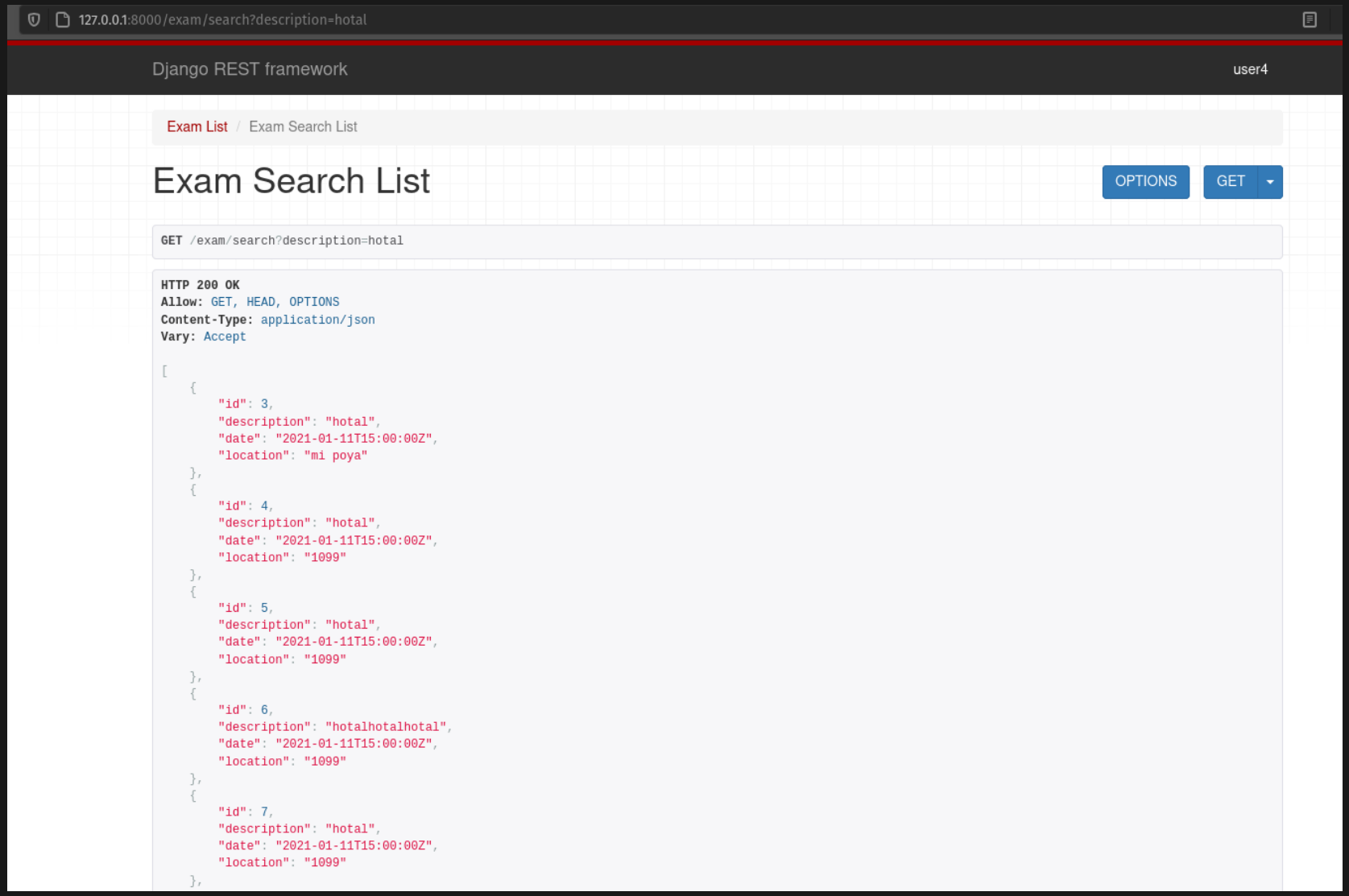


Figure 10: Search exam

4.3 GRADES

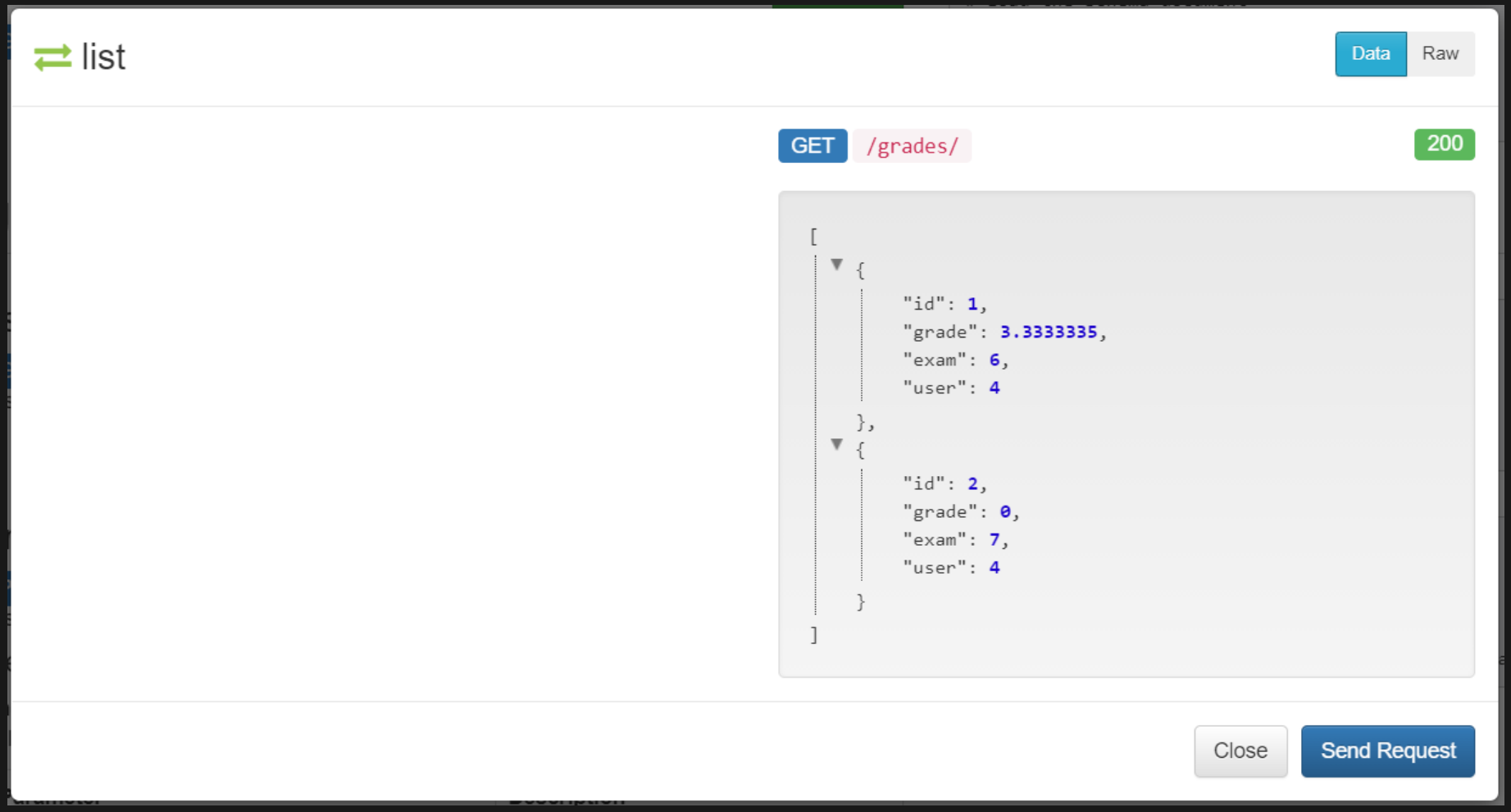


Figure 11: List grades

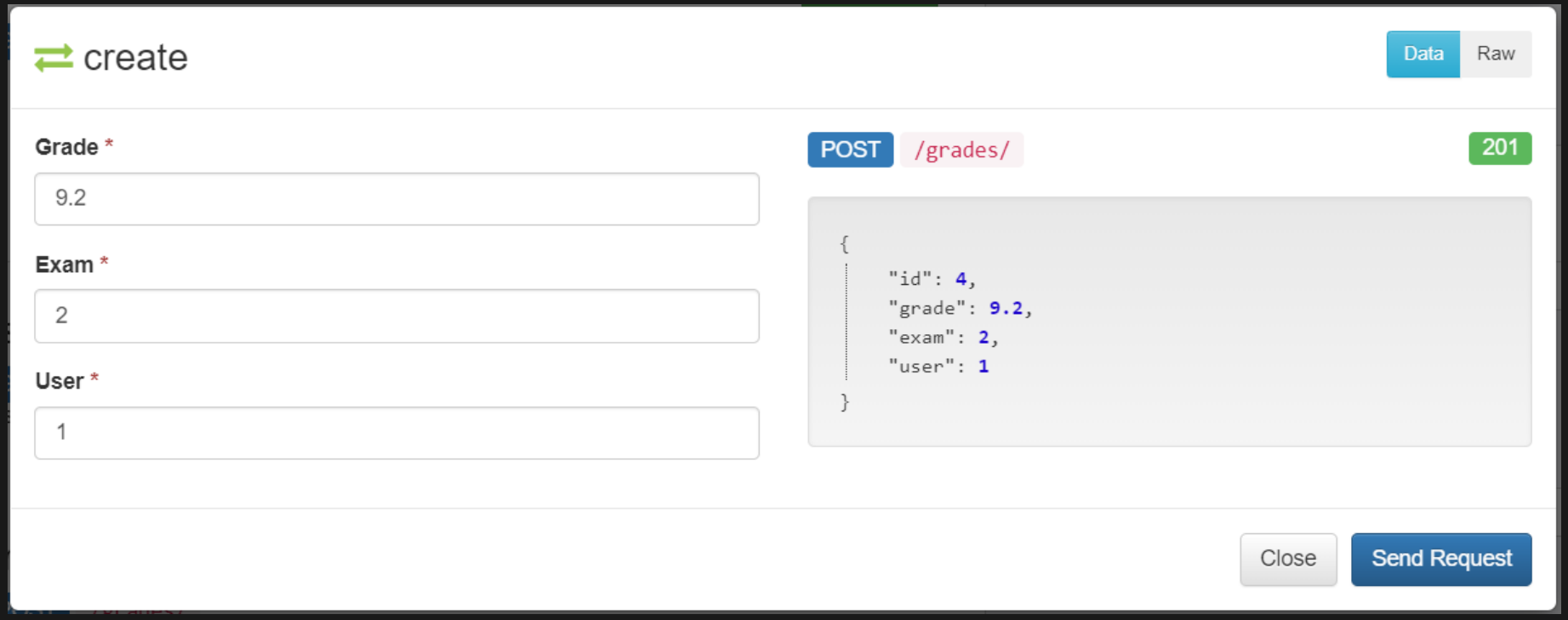


Figure 12: Create grade

Figure 13: Read grade

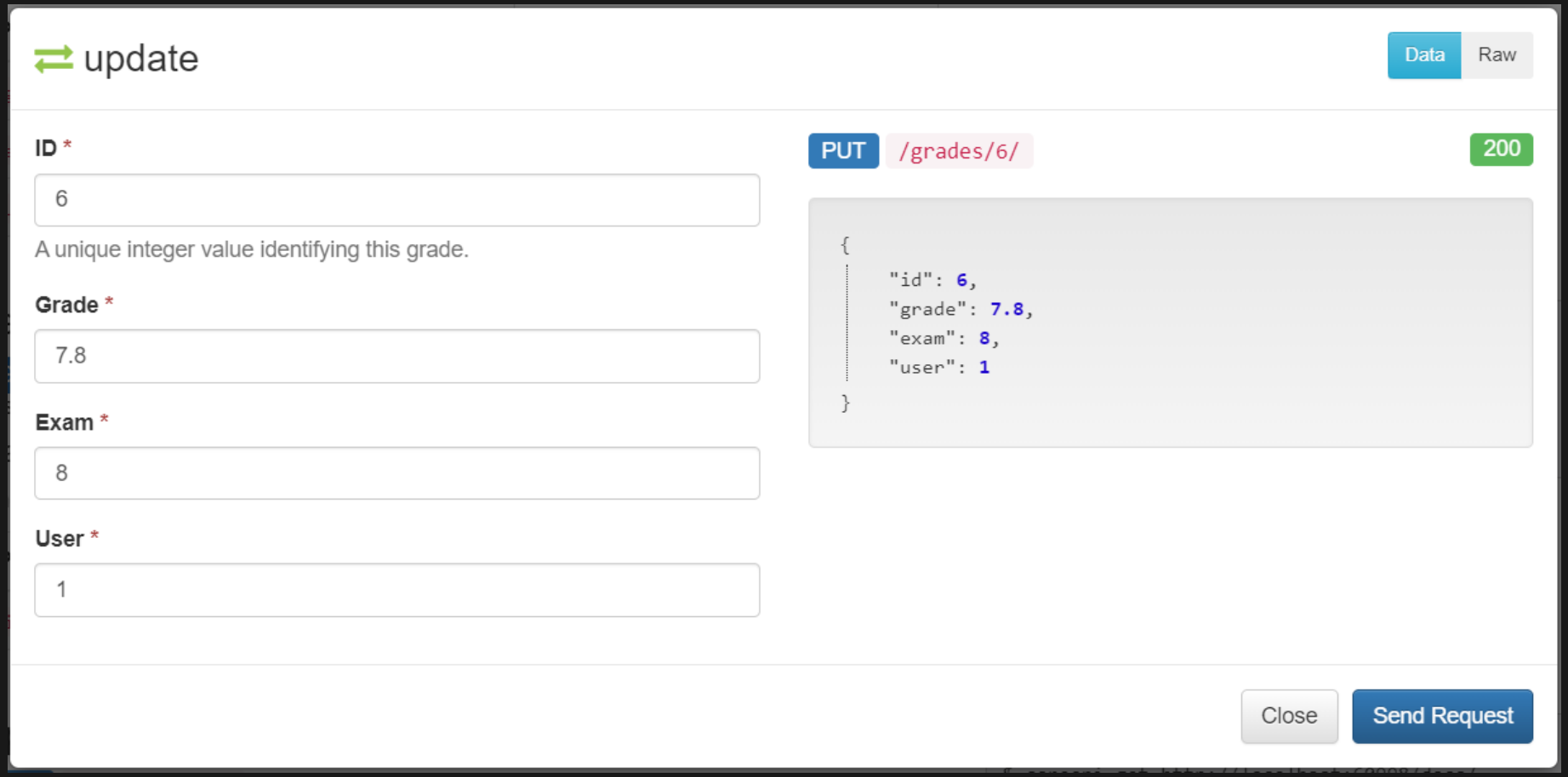


Figure 14: Update grade

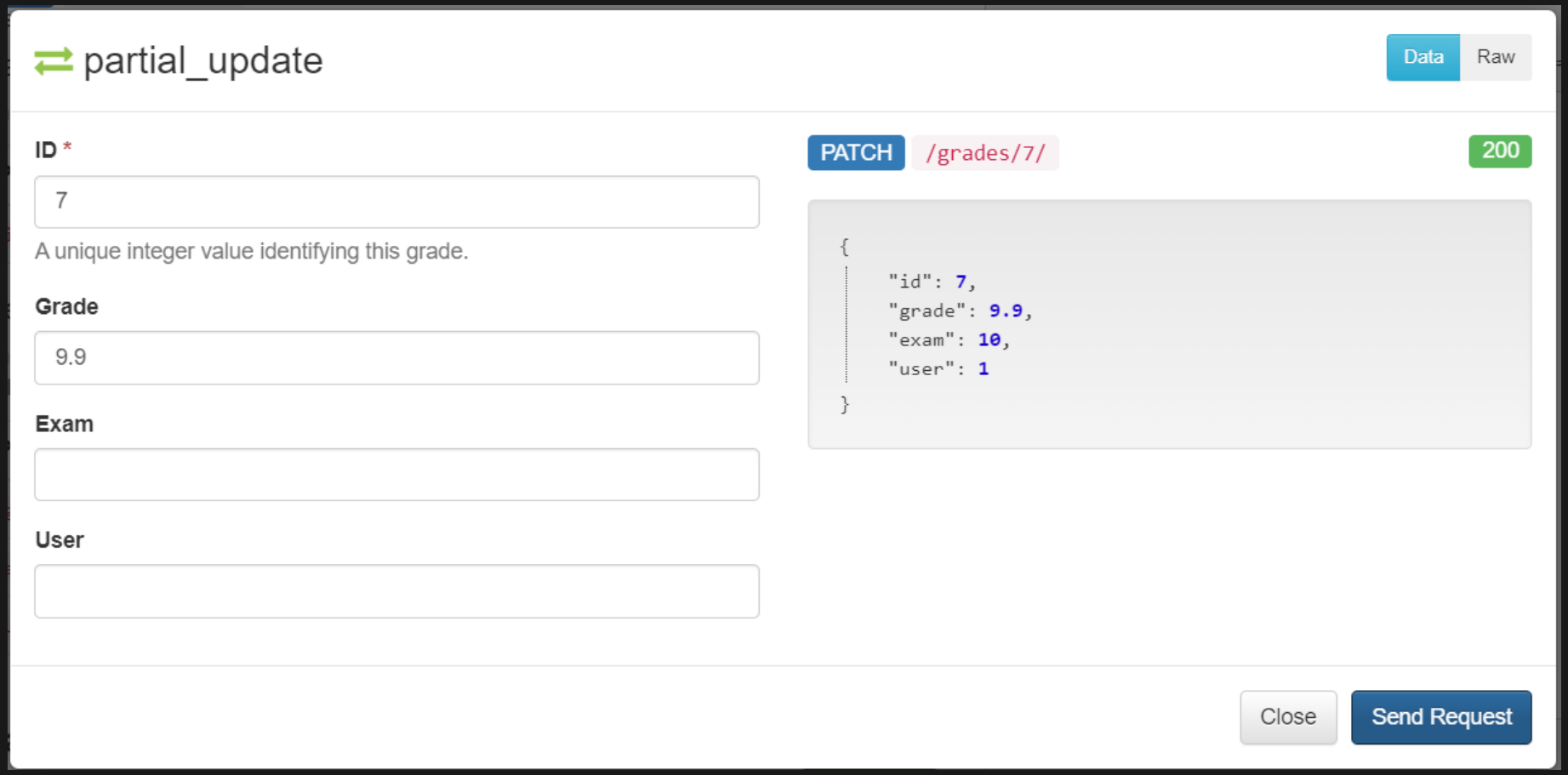


Figure 15: Patch grade

Figure 16: Delete grade

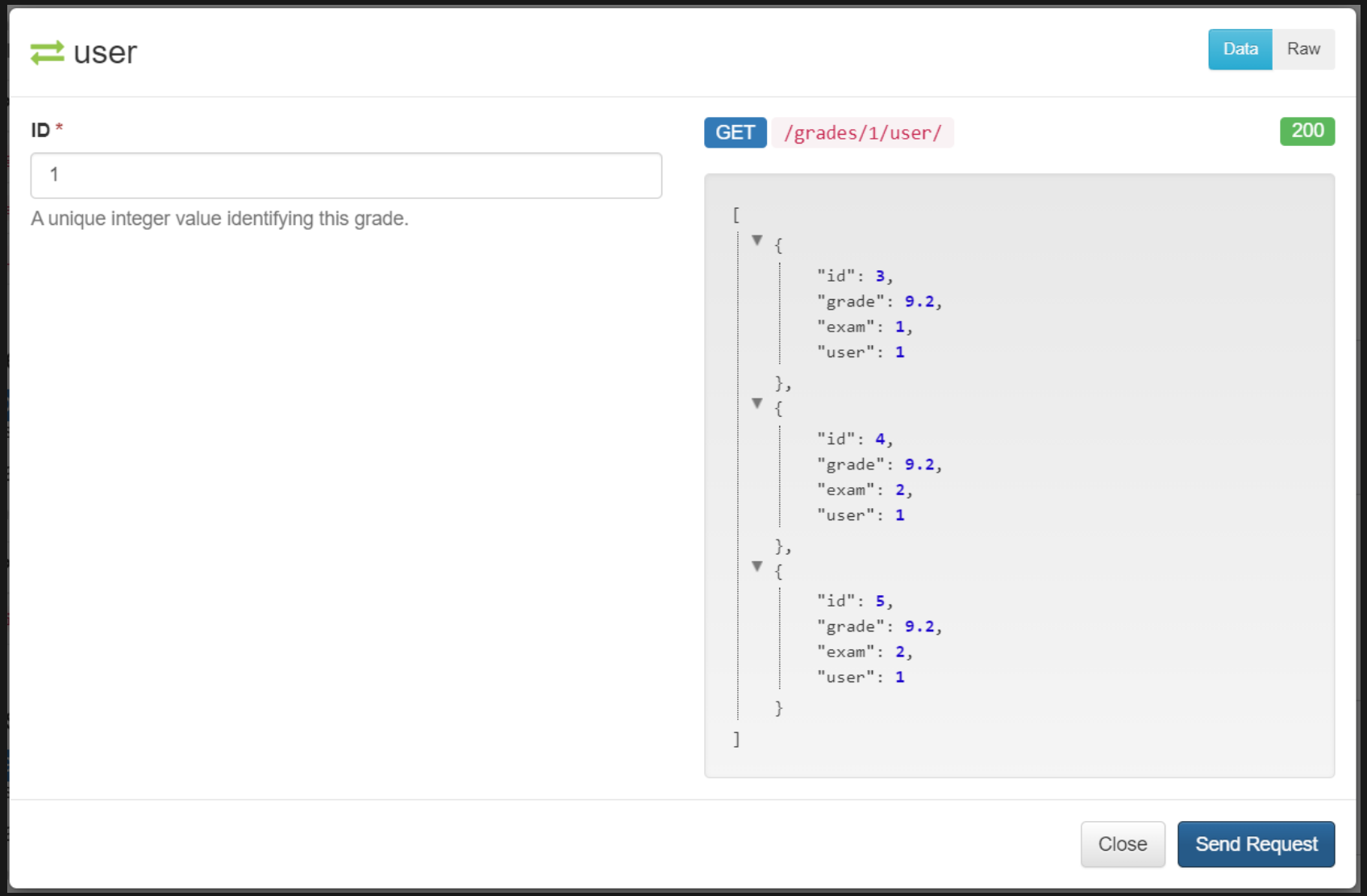


Figure 17: Search user grades

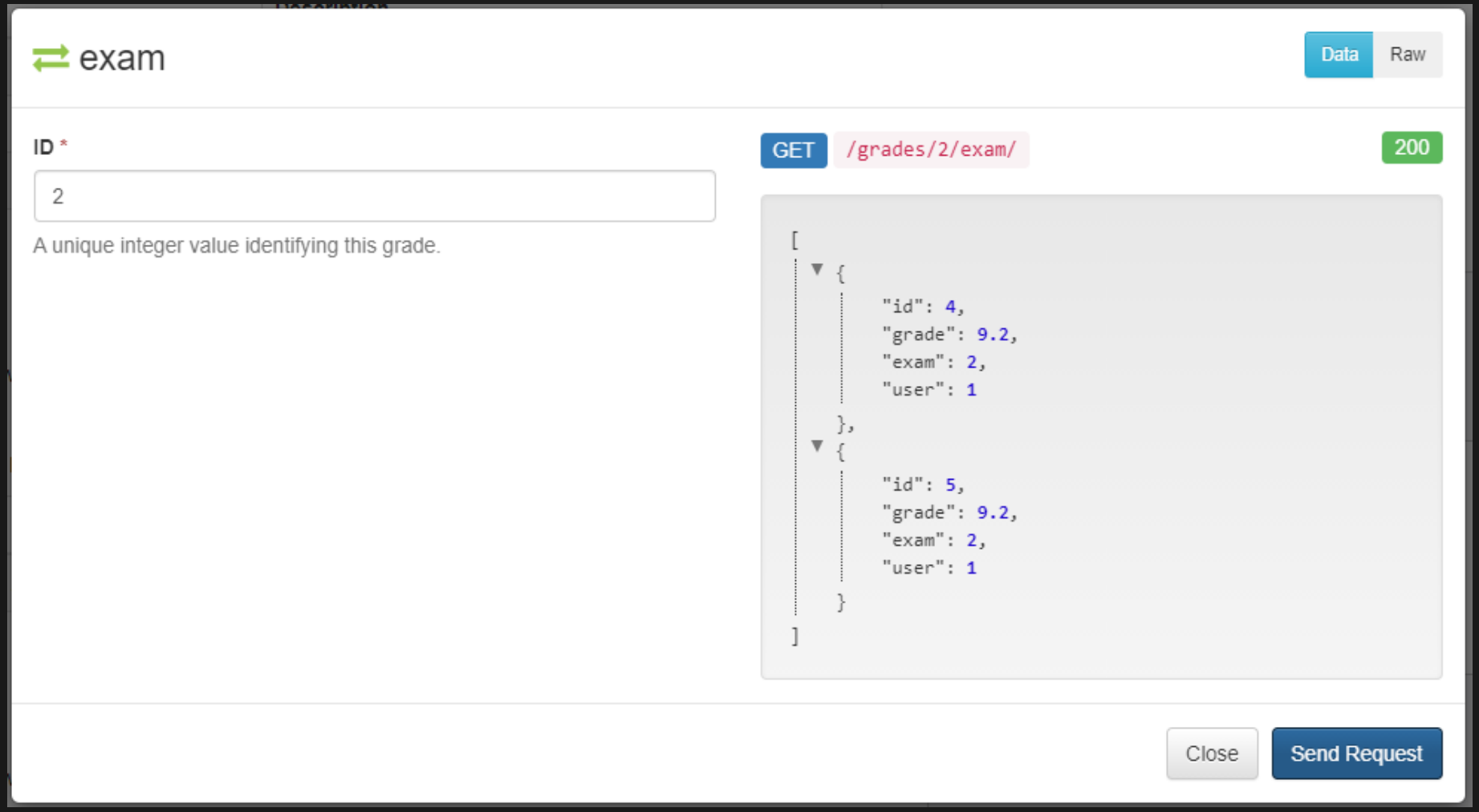


Figure 18: Search exam grades

5 SOLUTION JUSTIFICATION

5.1 WEB SERVICE

5.1.1 TECHNOLOGIES

5.1.2 VIEWSETS AND GENERICS

5.1.3 DECISIONS: AUTHENTICATION

5.2 RMI MODIFICATIONS

5.2.1 HTPP

5.2.2 CLIENT FLOW CHANGES

1. Search
2. List
3. Choose

5.2.3 SERVER FLOW CHANGES

1. Description
2. Date
3. Location

5.3 TIME DEDICATED

It is difficult to say, but we estimate an approximation of 90 hours. We are a group of three students, and we worked in this project for 6 days, 5 hours each day.