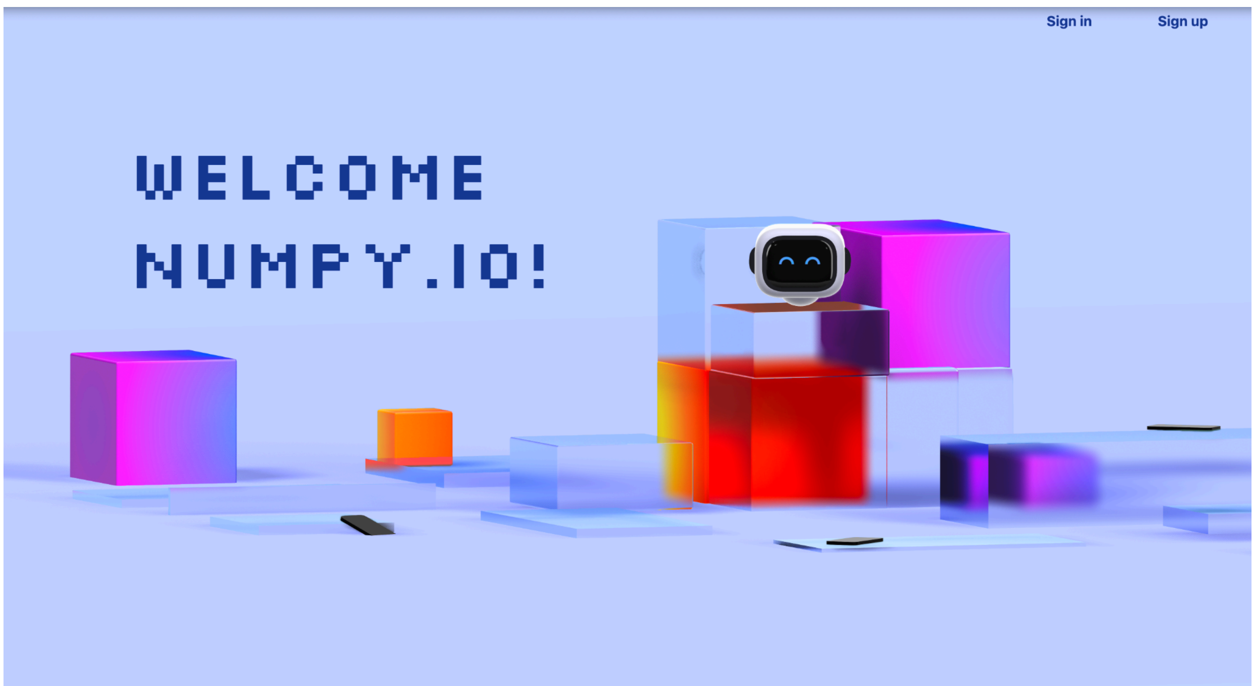
Natasha Piedrabuena
Numerical Computation

Final Project

In this final project we develop and implement a web application software system for two key numerical methods: differentiation and integration. These systems are designed to enhance computational accuracy and efficiency while providing users with an intuitive interface for solving complex mathematical problems.

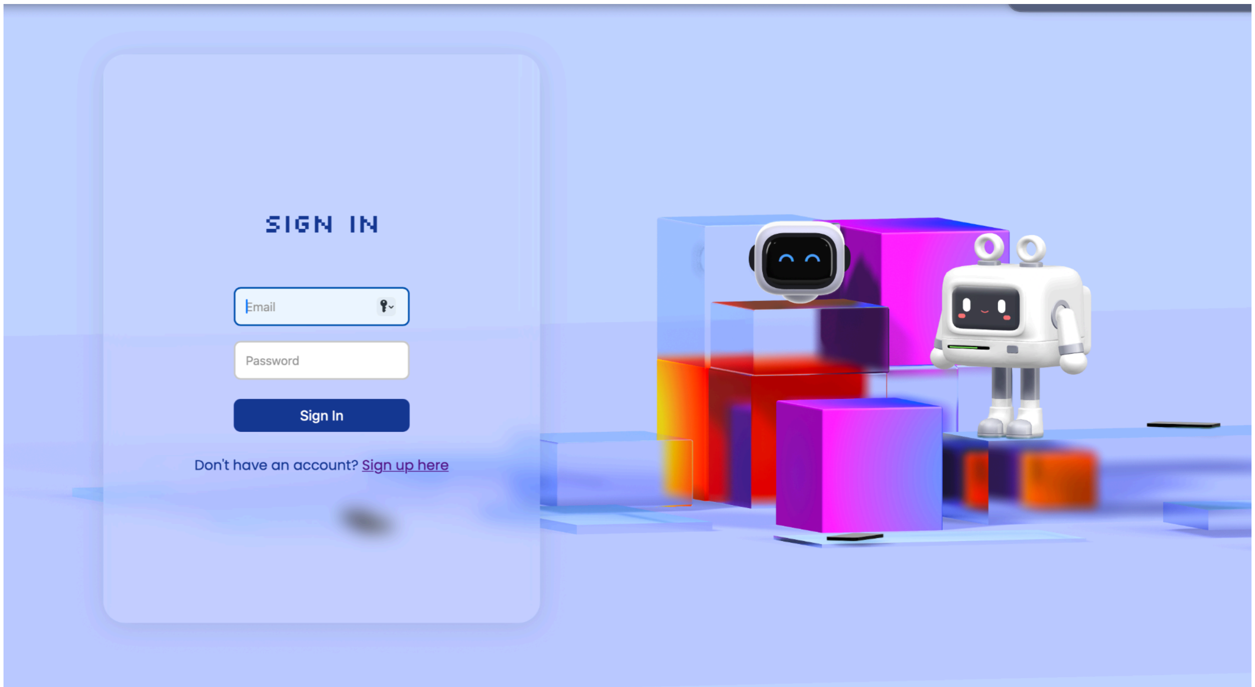
For our frontend web application we used a web application framework called React JS, for user Authentication we used firebase and for the backend we used Flask python server.

The experience starts off with a preloading animation that will give time for the 3d scene to render, it takes around 3 seconds and the animation takes 3 seconds. This is so the user does not need to wait for the scene to load through the hero section which will give the user a glitchy scene.

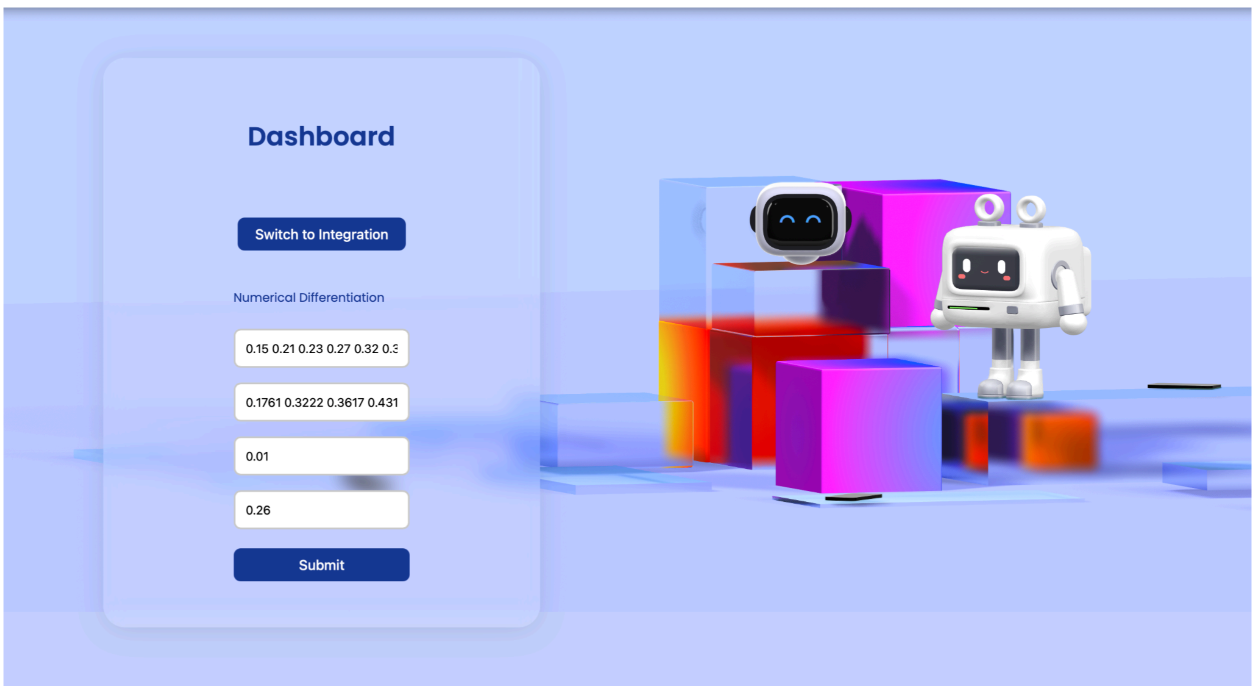


The user interface is interactive where you can interact with the robot and it sings. We also incorporated react sound so we can have music playing in the background when the user interacts with certain components.

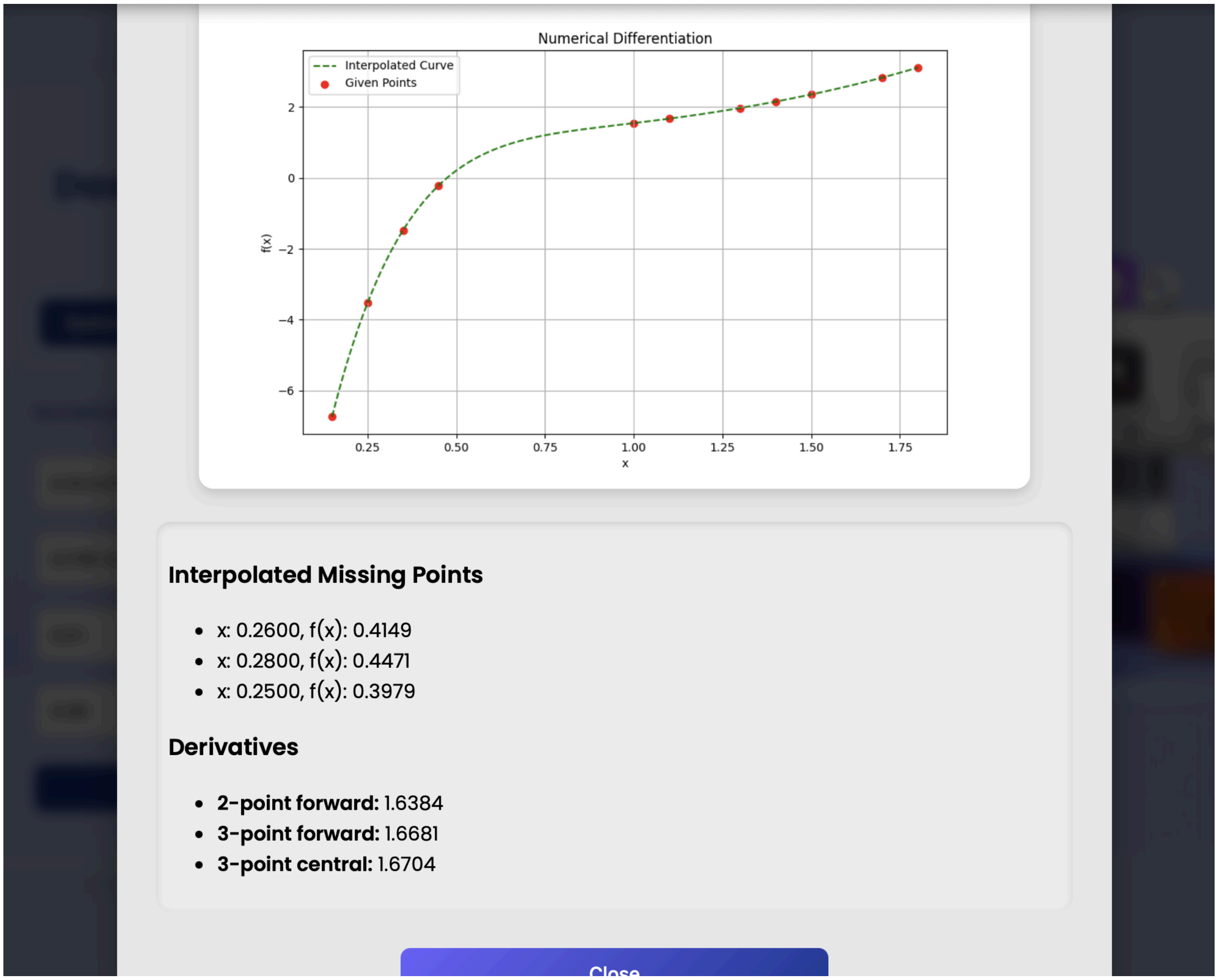
When the user initiate Sign in/ Sign up, the react router dom is initiated where it will navigate to the AuthForm.jsx, to interact with the sign in and sign up functions.



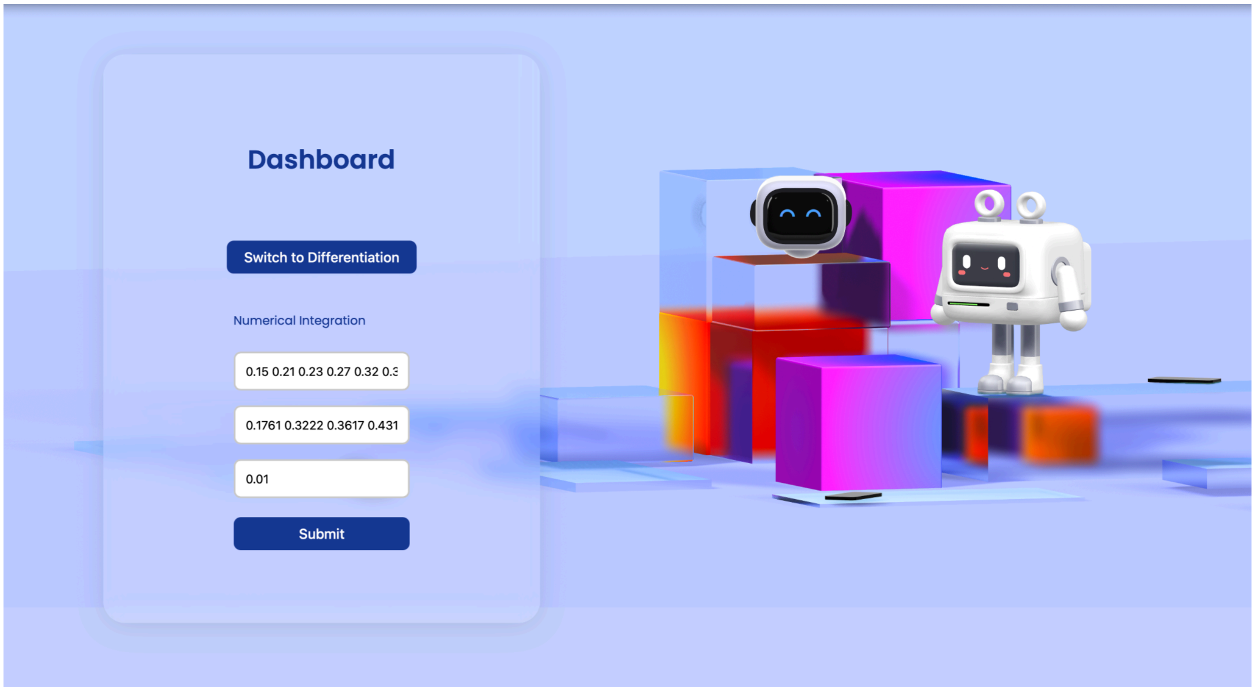
The authentication is handled by the firebase secure authentication system that developers can incorporate in their projects.



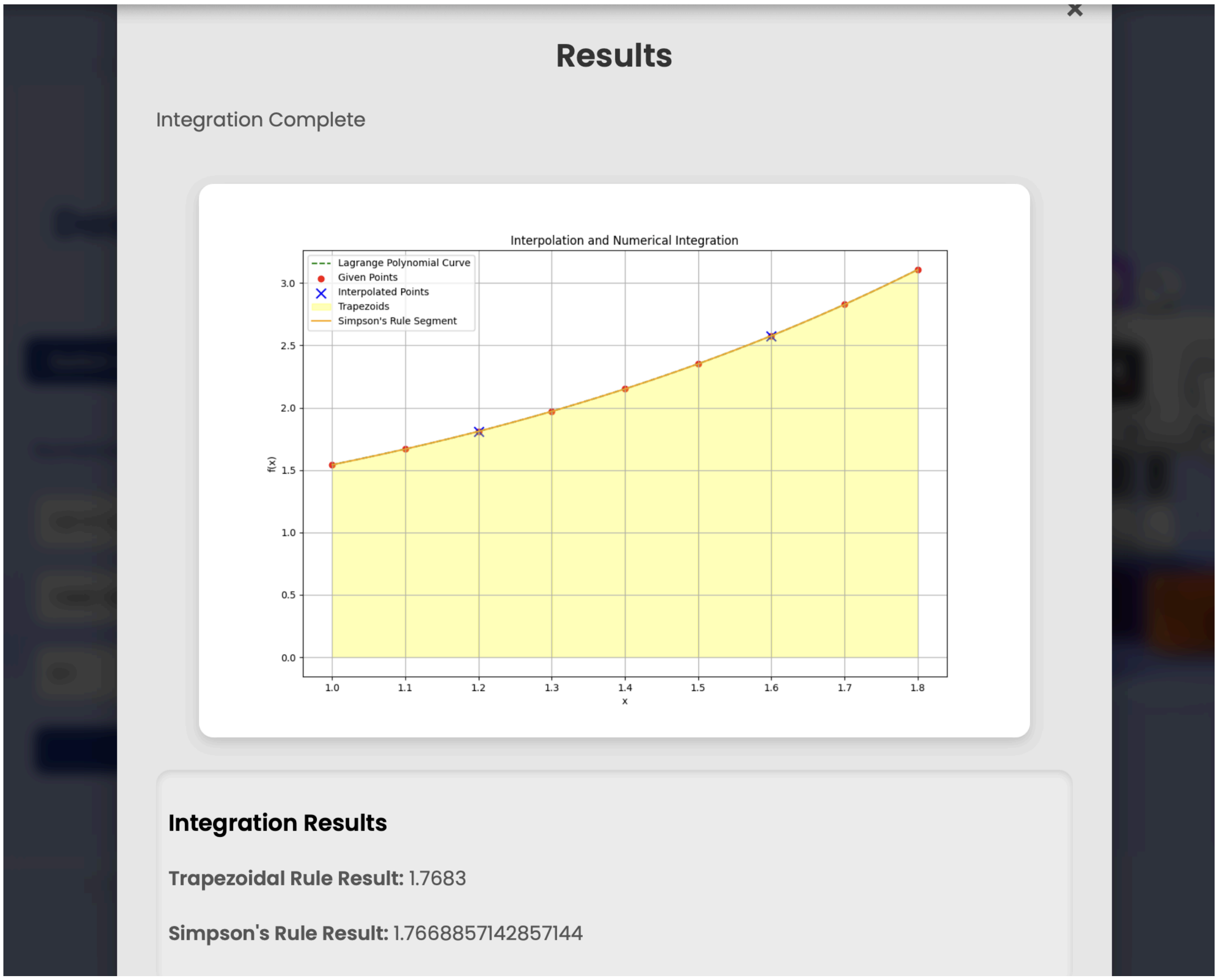
The numerical differentiation system allows users to calculate the derivative of a function at a specific point using three different approaches: the two-point forward difference formula, the three-point forward difference formula, and the three-point centered difference formula. These methods are foundational in numerical analysis, offering varying levels of precision and applicability based on the nature of the input data. To address incomplete or irregular data, the system incorporates Lagrange interpolation, a widely-used technique for estimating missing values in a dataset. This feature ensures that users can work with imperfect data without compromising the reliability of the results.



Users can either manually input the function values, select the differentiation method that best suits their needs, and input the point where the derivative is to be computed. The system then calculates and displays the derivative, providing an essential tool for mathematical analysis in scientific and engineering applications.
In this case displays the interpolated missing points + the derivatives for the 2 point forward, 3 point forward and 3-point center.



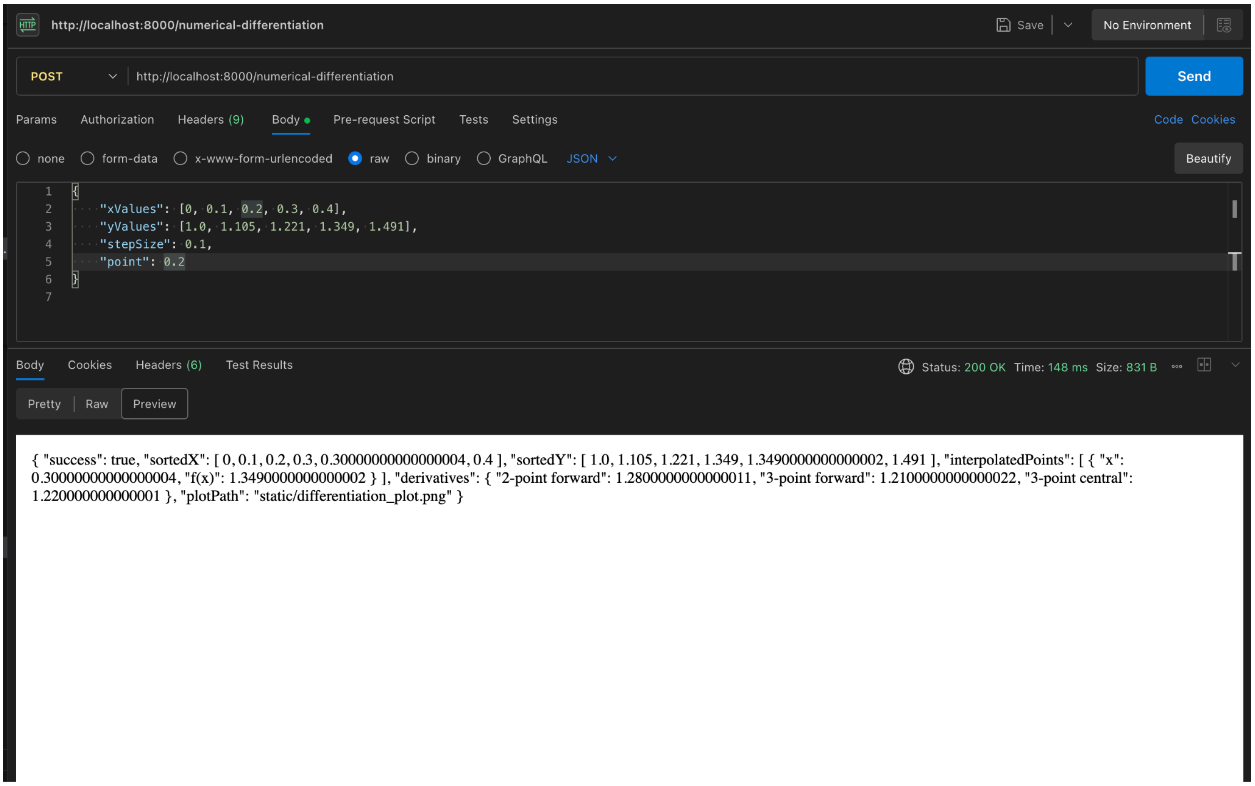
The numerical integration component of the project complements the differentiation system by focusing on the calculation of definite integrals for a given function. This system is built around two established numerical methods: the trapezoidal rule and the composite Simpson's rule. These methods are particularly effective for approximating the integral of a function over a specified interval, even when the function cannot be integrated analytically. The trapezoidal rule is known for its simplicity and versatility, while the composite Simpson's rule offers higher accuracy for smooth functions by using parabolic segments. Like the differentiation system, the integration tool also employs Lagrange interpolation to fill in missing data points, ensuring a seamless and accurate computational process.



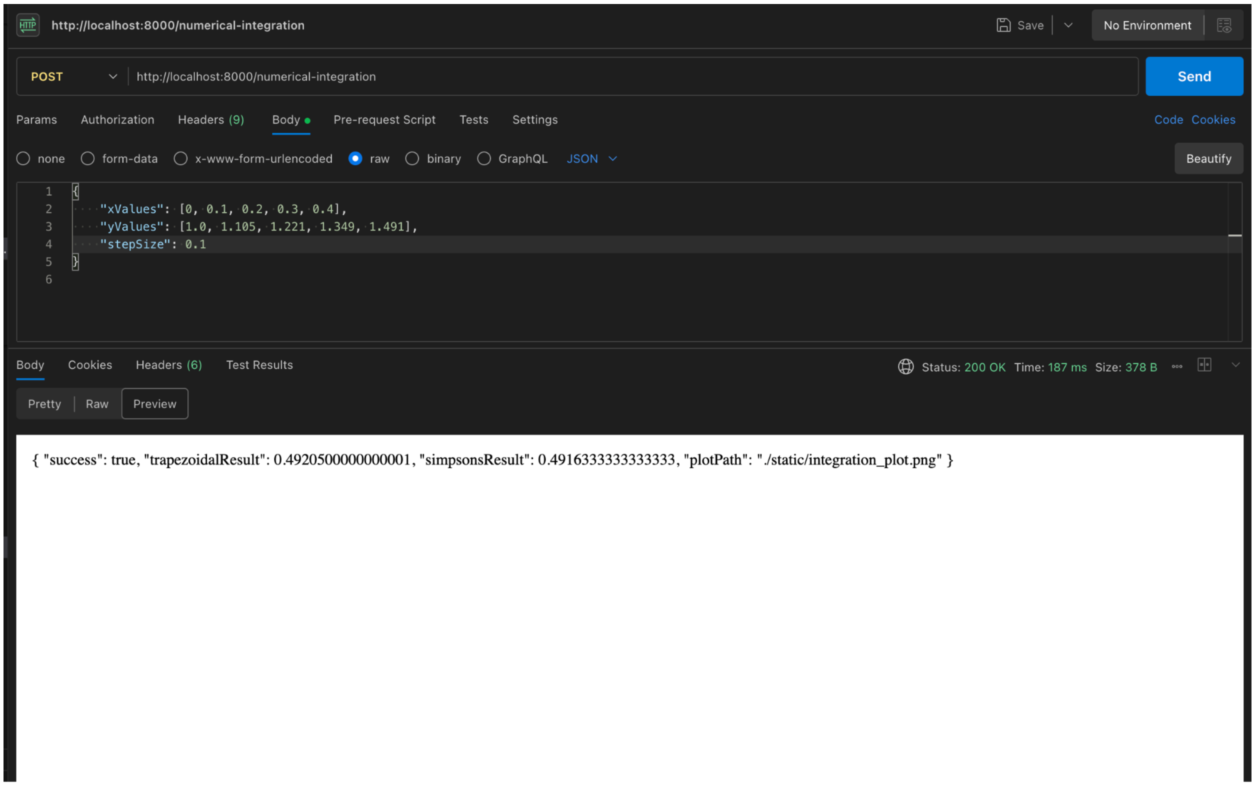
Users can enter or upload function data, specify key parameters such as the step size (h). The output is a precise numerical approximation of the integral, making the system suitable for a wide range of real-world applications, from physics and engineering to data analysis and financial modeling.

The main struggle for me in this project was the integration of the frontend and backend components. To ensure communication between the frontend and backend server, I used a postman to test the integration of the system endpoints while my partner worried about the backend for the integration implementation and the dashboard page.

In the post request we see a successful request and the data that is being returned from backend to the frontend where it will then be displayed to the user. Due to issues with the display component and overflow error we only displayed the interpolated missing points and the 2 forward, 3 point forward and 3 point central.



Endpoint for Numerical Differentiation



Endpoint for the Integration

I also did testing for the functions with scripts and I got success status for all my use cases.

Overall the goal of these systems is to combine mathematical rigor with user-friendly functionality, enabling both experts and novices to perform critical computations with ease. By providing clear input options and detailed outputs, the software bridges the gap between theoretical numerical methods and practical problem-solving. Furthermore, the integration of interpolation techniques ensures that the systems can handle real-world data imperfections, making them versatile tools for academic, professional, and industrial use. The design philosophy prioritizes accuracy, reliability, and ease of use, reflecting a commitment to creating software that meets the diverse needs of its users. Whether for educational purposes, research, or professional projects, these numerical differentiation and integration tools are poised to become indispensable assets in computational analysis.