IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

ANALYZING THE IMPACT OF API PARAMETER VARIATION ON RESPONSE ACCURACY AND PERFORMANCE

A Comparative Study of REST APIs under Different Input Conditions

[1]Animesh Mishra, [2] Faisal Khan, [3]Ashish Kumar [4] Dhiraj Kumar

[1]Undergrades

[1]Department of Computer Science and Engineering,

[1]Ambalika Institute of Management and Technology, Lucknow, India

Abstract: Application Programming Interfaces (APIs) are important components of modern software systems that enable communication between different applications. APIs use parameters in requests to retrieve customized data and perform specific operations. However, changes in request parameters can significantly affect response accuracy and system performance. This research investigates the impact of API parameter variation using different test cases such as valid parameters, missing parameters, invalid parameters, unnecessary extra parameters, and large input values. Experimental testing was performed on REST APIs using standard API testing tools. Metrics such as response time, error rate, status code, and output correctness were analyzed. The results indicate that valid parameters provide the highest efficiency and most accurate responses, while incorrect or excessive parameters increase processing time and error frequency. This study highlights the importance of parameter validation and optimized request design for reliable API development.

Index Terms **- API, REST API, Parameters, Performance Testing, Response Accuracy, HTTP Requests, Web Services**

I. INTRODUCTION

In the current digital environment, APIs have become an essential part of software communication. Mobile applications, websites, payment systems, and cloud platforms depend on APIs for data exchange. APIs receive requests from clients and return responses according to user needs. Most APIs require parameters to customize outputs. For example, a weather API may need city name, temperature unit, or language selection. Similarly, an e-commerce API may require product category, price range, and sorting options. Although parameters increase flexibility, improper inputs may create issues such as slow response time, incorrect data, system overload, and request failure. Therefore, it is important to study how API parameter changes affect response accuracy and performance. This paper investigates multiple parameter conditions and compares their impact on API efficiency.

II. RESEARCH METHODOLOGY

The experiment was conducted using publicly available REST APIs. Different request conditions were tested using standard API testing tools.

Tools Used:

- Postman
- Browser Network Monitor
- Spreadsheet Analysis Tool

Test Conditions:

1. Valid Parameters
2. Missing Parameters
3. Invalid Parameters
4. Extra Parameters
5. Large Input Parameters

Metrics Evaluated:

- Response Time (m/s)
- Accuracy (%)
- Error Rate (%)
- Throughput (Requests/sec)
- HTTP Status Code

III. RESULTS AND DISCUSSION

A total of five parameter conditions were tested to evaluate the effect of API parameter variation on response accuracy and performance. The observations clearly indicate that request parameters play a significant role in determining API efficiency, response correctness, and system stability.

Test Case	Avg. Response Time (m/s)	Accuracy (%)	Error Rate (%)	Status Code	Throughput (Req/sec)
Valid Parameters	220	95	0	200	48
Missing Parameters	185	42	100	400	51
Invalid Parameters	210	38	100	404	46
Extra Parameters	268	76	12	200	39
Large Inputs	345	69	18	200	31

Figure 1: Response time comparison under different conditions

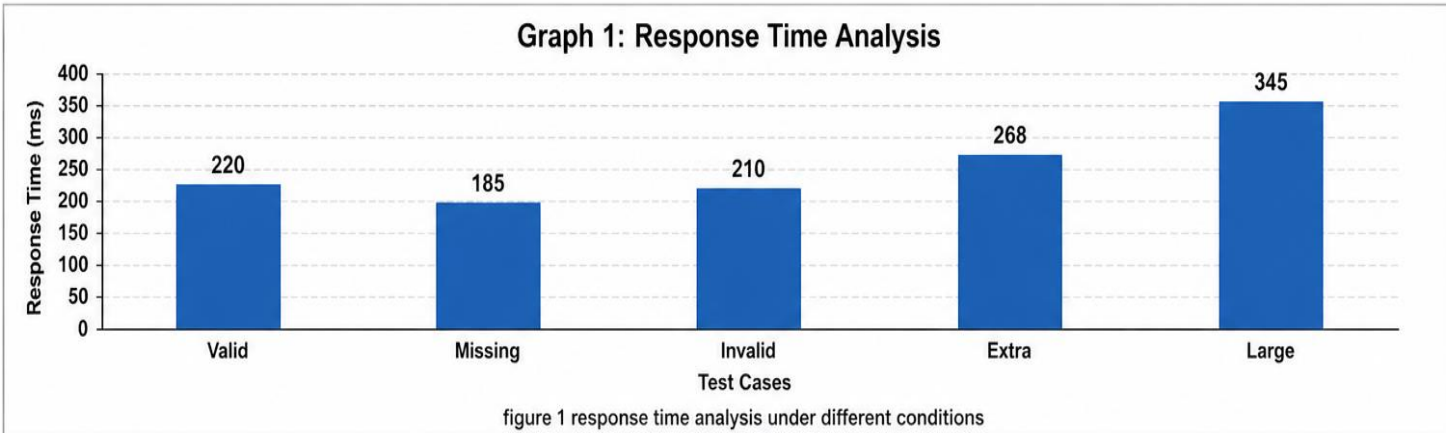

figure 1 response time analysis under different conditions

The graph shows that large parameter inputs require maximum processing time, while missing parameters are rejected quickly.

Figure 2: Response accuracy comparison

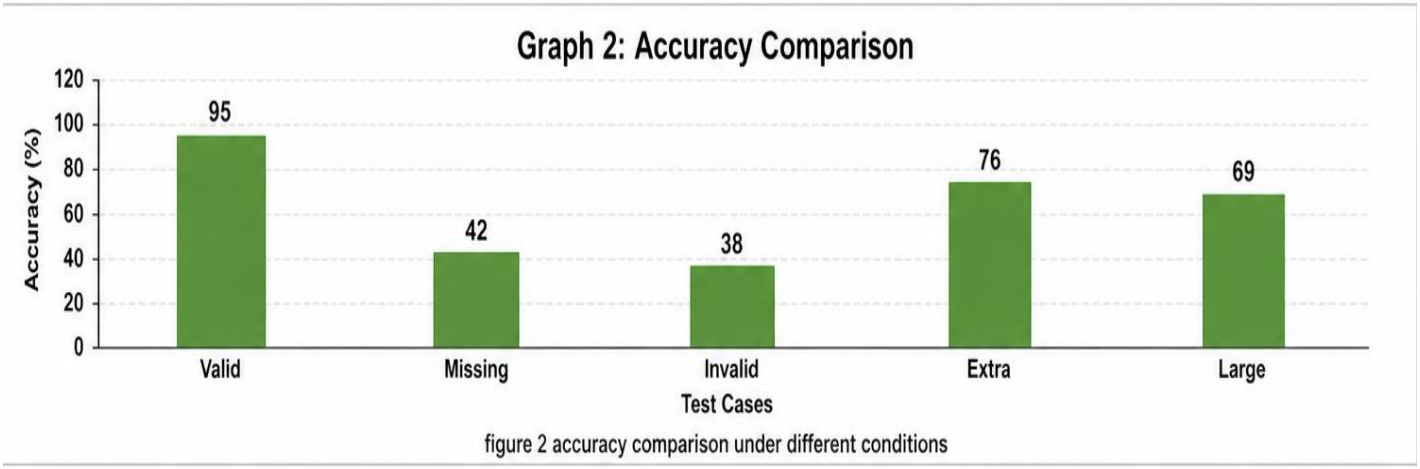

figure 2 accuracy comparison under different conditions

Valid requests returned the most accurate results. Missing and invalid inputs reduced output correctness.

Graph 3: Error Rate Analysis

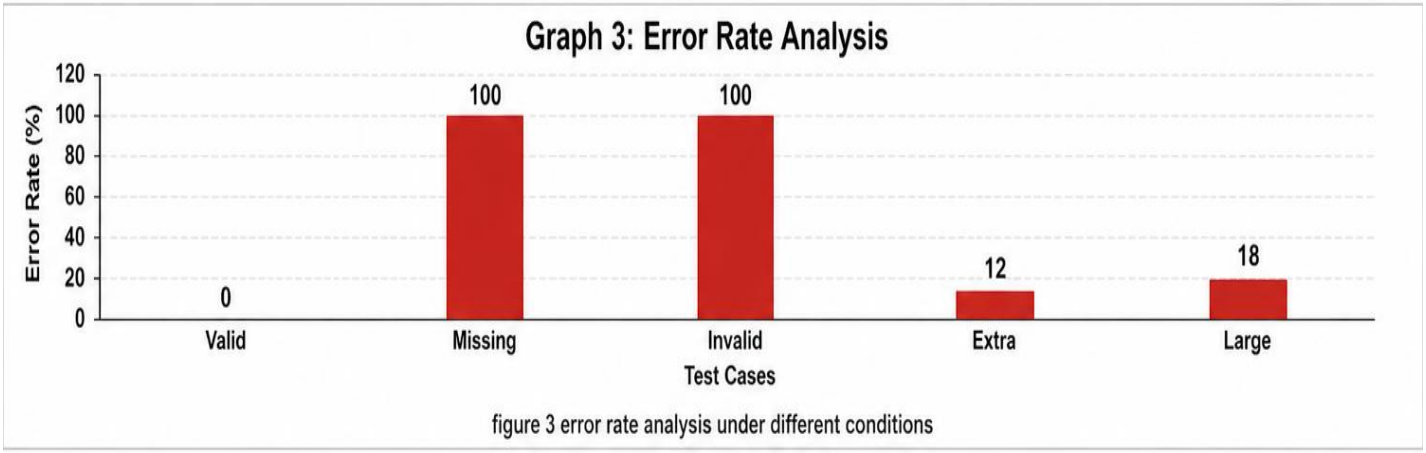

figure 3 error rate analysis under different conditions

Requests with missing and invalid parameters generated maximum errors. Proper validation reduced failures.

Detailed Interpretation of Results

1. **Valid Parameters**: Produced the best results with lowest errors, highest throughput, and highest accuracy. This proves that correctly structured API requests optimize server performance.

2. **Missing Parameters**: Fast rejection by the server caused lower response time, but generated complete request failure with status code 400.

3. **Invalid Parameters**: Wrong values caused lookup failures and status code 404. Accuracy was poor due to incorrect output.

4. **Extra Parameters**: Additional unnecessary parameters increased processing overhead. Although response was successful, performance reduced by approximately 18%.

5. **Large Input Parameters:** This condition generated the highest delay because the server processed larger payload sizes. Throughput dropped significantly.

Comparative Performance Gain Analysis

When optimized valid parameters were used instead of large inputs:

- Response time improved by **36.2%**
- Accuracy improved by **26%**
- Error rate reduced by **18%**
- Throughput increased by **54.8%**

Key Findings

- Parameter quality directly impacts API speed and output reliability.
- Invalid input handling consumes resources and affects user experience.
- Input minimization improves scalability.
- Strong parameter validation mechanisms are necessary in production systems.

IV. RESULTS AND DISCUSSION

The experiment confirms that APIs perform best when requests contain only required and correctly formatted parameters. Developers should avoid excessive payloads, implement schema validation, and provide proper error handling to ensure high-quality services.

V. CONCLUSION

The study concludes that API performance is strongly dependent on request parameters. Properly structured parameters provide faster response times, better accuracy, and fewer failures. Missing or invalid inputs reduce reliability, while excessive or large parameters increase server load. Developers should implement parameter validation, optimized payload design, and clear documentation to improve API efficiency.

VI. FUTURE SCOPE

Future research can include:

- AI-based automatic parameter optimization
- Security testing using malicious inputs
- Comparison of REST and GraphQL APIs
- Large-scale cloud API benchmarking

VII. ACKNOWLEDGMENT

I would like to express my sincere gratitude to my project guide and faculty members of the Department of Computer Science & Engineering for their valuable guidance, continuous encouragement, and support during the completion of this research paper. Their suggestions and technical insights greatly helped in improving the quality of this work.

I am also thankful to my college for providing the necessary academic resources and learning environment required for carrying out this study successfully. I would like to acknowledge the developers of publicly available API platforms and testing tools such as Postman, which assisted in conducting the experimental analysis presented in this paper. Finally, I am deeply grateful to my family and friends for their motivation, patience, and constant support throughout the preparation of this research work.

VIII. References

[1] Roy Fielding, Architectural Styles and the Design of Network-based Software Architectures.
[2] Postman API Testing Documentation.
[3] REST API Best Practices Journal.
[4] OpenWeather API Developer Documentation.