Proceeding Series of the Brazilian Society of Computational and Applied Mathematics, v. 12, n. 1, 2026.

Trabalho apresentado no XLIV CNMAC, Fundação Getulio Vargas - Rio de Janeiro - RJ, 2025.

Proceeding Series of the Brazilian Society of Computational and Applied Mathematics

Efficient Lupus Capturing Data System Based on Model View Controller

Leticia Mendieta[1], Christian E. Schaerer[2], Santiago Gomez[3], Oliver Medina[4]
Polytechnic School, UNA, Paraguay
Isabel Acosta[5], Francisco Santacruz[6], Juana Ortellado[7], Gladys Velázquez[8], Clyde Parodi[9], Astrid Paats[10]
Medical School, UNA, Paraguay

Systemic lupus erythematosus (SLE) is a chronic autoimmune disease that presents several challenges in research and patient follow-up. One of the main issues is the overlap with regular consultations, which leads to a lack of comprehensive clinical databases. Traditional data capture methods often lack the necessary flexibility and efficiency, resulting in discrepancies that complicate statistical analysis and, in some cases, may invalidate it. This makes it extremely difficult to predict disease progression. To address these challenges, this work explores a robust data capture system grounded in mathematical principles. Such a system is focused on statistical analysis, developing predictive models, and classifying cases requiring early and differentiated care.

This work presents a data capture system based on the Model-View-Controller (MVC) design pattern, commonly used in web development[2]. MVC promotes better organization and code management, simplifying maintenance and future updates. It separates responsibilities: the model handles data logic, the view manages presentation, and the controller coordinates interaction. This structure enhances code reuse, simplifies maintenance, and facilitates the addition of new features. It is especially useful for monitoring patients with SLE. Figure 1 shows the system's interconnection scheme.

The MVC-based (LUPUS-Capturing system) efficiently organizes and manages data while optimizing request response times. This is particularly crucial for large volume data applications. Response time is calculated using the standard formula [3]: $T_r = T_w + T_p + T_c$, where T_r is the response time, T_w is the waiting time, T_p is the processing time, T_c is the transmission or communication time. Table 1 shows the response times of the processed requests, which are broken down into waiting, processing, and transmission times.

[1]lcmendieta@pol.una.py
[2]cschaer@pol.una.py
[3]sgomez@pol.una.py
[4]olomedina00@gmail.com
[5]dr.acostacolman@gmail.com
[6]fsantas@gmail.com
[7]juaniorte@yahoo.com.mx
[8]velazquez.aguayo@gmail.com
[9]parodiclyd@hotmail.com
[10]astrid.paats@gmail.com

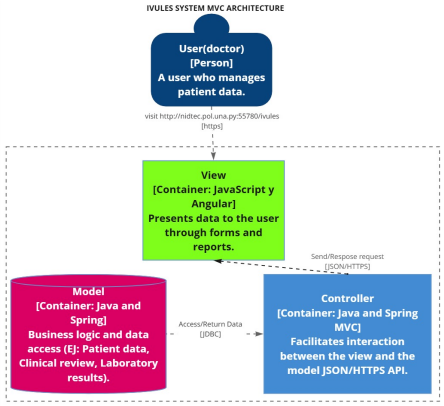

Figure 1: IVULES system MVC interactions. Source: authors.

Table 1: Response time (following [1] criteria).

Request	T_w(ms)	T_p(ms)	T_c(ms)	T_r(ms)
1	0.079	237.24	0.30	237.62
2	0.10	246.01	0.52	246.63
3	0.085	245.59	0.96	246.61
4	0.37	245.18	0.95	246.50

Response times are observed to be consistent and stable, with low variations in the mean 244.86 ms. Processing is the most significant time consuming (approx. 243.94 ms), while waiting time and transmission time are minimal (approx. in mean 0.159 ms and 0.6825 ms, respectively). Hence, lattice time is not a significant problem. Using the MVC pattern in system development has proven to be an optimal solution to the challenges of monitoring patients with systemic lupus erythematosus (SLE). Efficient response times and flexibility allow continuous system improvement, facilitating SLE patient monitoring and supporting medical research.

Acknowledgment

Authors acknowledge project PROCIENCIA-CONACYT-PROCIENCIA PINV01-685.

References

[1] R. V. Rochim, A. Rahmatulloh, R. R. El-Akbar, and R. Rizal. “Performance Comparison of Response Time Native, Mobile and Progressive Web Application Technology”. In: **Innovation in Research of Informatics (INNOVATICS)** 5 (May 2023). DOI: 10.37058/innovatics.v5i1.7045.

[2] H. Wang, N. Wang, M. Li, S. Mi, and Y. Shi. “Student Physical Health Information Management Model under Big Data Environment”. In: **Scientific Programming** 2021 (July 2021), pp. 1–10. DOI: 10.1155/2021/5795884.

[3] J. Yang and A. Abraham. “Analyzing the Features, Usability, and Performance of Deploying a Containerized Mobile Web Application on Serverless Cloud Platforms”. In: **Future Internet** 16 (Dec. 2024), p. 475. DOI: 10.3390/fi16120475.