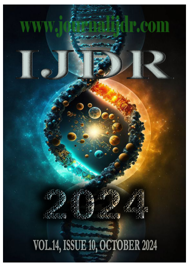
ISSN: 2230-9926

Available online at http://www.journalijdr.com

IJDR

International Journal of Development Research
Vol. 14, Issue, 10, pp. 66847-66852, October, 2024
https://doi.org/10.37118/ijdr.28866.10.2024



RESEARCH ARTICLE OPEN ACCESS

DEVELOPMENT OF A WEB APPLICATION FOR PROCESS AUTOMATION FOR THE VETERINARY PATHOLOGY SECTOR OF CENTRO UNIVERSITÁRIO DO ESPÍRITO SANTO

Lucas Faé Baldan*[1] and Allan Stieg Candeia[2]

[1]Graduando em Sistemas de Informação, Centro Universitário do Espírito Santo – UNESC, Espírito Santo, Brasil
[1]Mestre em Tecnologias Emergentes na Educação, MUST University, Flórida, USA

ARTICLE INFO

ArticleHistory:
Received 17th July, 2024
Received in revised form
03rd August, 2024
Accepted 11th September, 2024
Published online 30th October, 2024

Key Words:
Data, Management, Pathology, Systems, Veterinary, Web.

****Corresponding Author: Lucas Faé Baldan,***

ABSTRACT

The article discusses the development of a web application for the veterinary pathology department at the University Centre of Espírito Santo, motivated by the need to modernize and automate the process of managing diagnostic records. The aim was to create a digital system that would replace manual methods, providing greater agility and precision in the management of information. To this end, applied research methodologies were employed, with meetings held with professionals in the field of Veterinary Medicine to align the requirements and validate the functionalities developed. In addition, bibliographic research provided the necessary theoretical background. During development, specific business rules were implemented, along with the user interface, registration routines and procedures for analyzing data and issuing management reports. Robust security measures were adopted to protect the data, and internal procedures were improved to guarantee the integrity of the information. Unit and integration tests ensured the quality of the system, which is based on solid theoretical analysis. The result was an efficient and secure system, capable of meeting the needs of the sector's records control, making it easier to register, consult and issue diagnoses. In addition to optimizing internal processes, the solution enables future complex data analysis, contributing significantly to the quality of the services provided by the institution's pathology laboratory.

Citation*: *Lucas Faé Baldan and Allan Stieg Candeia. 2024. *"Development of a web application for process automation for the veterinary pathology sector of Centro Universitário do espírito Santo". International Journal of Development Research*, 14, (10), 66847-66852.

INTRODUCTION

Efficient information management is a crucial area in various sectors, including the fields of veterinary medicine and especially Information Systems, where the accuracy and accessibility of data is essential to guarantee the quality of the services provided. With technological developments, many institutions are looking to modernize their management, replacing manual and decentralized processes with integrated digital systems, making their day-to-day decision-making much more efficient. This movement is particularly relevant in veterinary hospitals, where the handling of pathological files can directly influence the effectiveness of treatments and clinical decision-making. At Centro Universitário do Espírito Santo's veterinary hospital, traditional methods of recording and archiving pathological care are still prevalent, using physical sheets and decentralized platforms to store information. These methods present several challenges, such as difficulties in searching and consulting data, the risk of records being lost or damaged, as well as readability problems. The implementation of a modern web application aims to resolve these issues by centralizing care records, which will potentially improve efficiency and information security.

The aim of this research was to create a digital system that would replace manual methods, providing greater agility and precision in the handling of information. The solution will replace outdated methods, improving the accuracy and accessibility of information, automating processes that currently consume significant time and wear out professionals in the field. To achieve this goal, the research adopts an exploratory methodological approach, combining applied and bibliographic research methods, with an emphasis on a case study and qualitative collection of results. The process began with interviews with professionals at the veterinary hospital, with the aim of understanding their needs, difficulties with current processes and the functioning of work routines and pathological studies. The solution was then developed using agile methodologies, integrating modern technologies and implementing strict access and security controls. Unit, integration and usability tests were conducted to guarantee the system's functionality and efficiency before its final implementation. The relevance of this research lies in the potential improvement of the services offered by the veterinary hospital, providing more effective and secure management of pathological care data. As well as directly benefiting the professionals and the animals treated, the solution will serve as a model for other institutions, promoting modernization and the adoption of advanced technological practices in the veterinary field. The centralization of records will reduce errors and data loss, resulting in a more reliable and efficient service for animal owners, while the training of more skilled professionals in digital tools will contribute to the continuous improvement of veterinary care.

MATERIALS AND METHODS

To carry out this research, two main methodologies were adopted: applied research and bibliographic research. When applied together, these methodologies provide procedures and techniques aimed at understanding the real challenges faced by professionals at the veterinary hospital. This combined approach allows for a practical and objective understanding of the sector's specific needs. At the same time, these methodologies served as a foundation for theoretical grounding through the consultation of scientific articles and books authored by recognized experts in the field. This step is essential for analyzing and comparing the results obtained from other studies, enabling the adoption of the best practices and techniques in the development of the proposed solution.

Applied research: Applied research focuses on problems that occur in the activities of institutions, organizations, groups or social actors. Its aim is to make diagnoses, identify problems and seek solutions. It responds to demands made by clients, social actors or institutions. It is characterized as an interventionist method, in which the researcher tests hypotheses related to the phenomenon of interest by implementing and evaluating changes in the real scenario. In this type of research, the researcher assumes responsibility not only for generating knowledge to help those involved, but also for applying it directly (Thiollent, 2009). The implementation of the applied research took place through a series of alignment meetings with the professionals in the institution's veterinary pathology department, which were fundamental to understanding the real problem faced by the team and mapping out the department's workflow. During these discussions, it was possible to gain a detailed understanding of the entire process of registering pathology records, from receiving the material to be analyzed, to filling in the records, to the final filing of the dossier, its printing and subsequent distribution. Throughout development, these meetings took place continuously, allowing the system to be adjusted as new needs or challenges were identified. This constant dialog was also crucial to ensure that the project did not deviate from the scope originally planned, ensuring that the solution efficiently met the expectations and demands of the professionals involved. After these meetings, all the processes identified were documented and analyzed in detail. In some cases, a re-analysis was necessary to refine the modeling and specification of the system's requirements. These processes served as the basis for drawing up the platform's functional and non-functional requirements, such as the integrity, security and reliability of the information entrusted to the system's management.

Bibliographic research: According to Boccato (2006), bibliographic research involves collecting and critical analysis of published documents on the topic under study, with the aim of updating and expanding knowledge, in addition to contributing to the research. With the theme defined and delimited, the researcher must follow paths that allow its development. Bibliographic research is based on books, theses, articles and other documents that assist in the investigation of the proposed problem. It is not enough to carry out a superficial review; it is necessary that the selected knowledge is significant and contributes to the advancement of the work. Literature research was essential to provide the theoretical basis necessary for the development of the platform; it involved the analysis of books and scientific articles related to pathological processes in veterinary medicine and systems automation technologies. This stage was carried out with the aim of supporting the technical and methodological decisions adopted throughout the project, in addition to identifying the best practices of data security and integrity standards, which are crucial in the development of systems involving sensitive information. This theoretical investigation also made it possible to align the proposed solutions with quality and security standards, ensuring that the platform was not only in compliance with the sector's best practices, but also prepared for future technological updates.

Technologies and tools used in Software Development: According to Miletto and Bertagnolli (2014), developing a software system is not a simple task, as it requires deep analysis and understanding of the problem to be solved. When directing software development toward a web platform, several aspects must be incorporated to ensure that the system is remotely accessible and secure through browsers. It is important to recognize how websites can be classified based on their functions and user interaction:

Site de conteúdo*:* apenas exibe um conteúdo de uma determinada área de forma linear, incluindo recursos (imagens, tabelas, gráficos etc.) e navegação simples.

Site de registro de dados ou entrada do usuário*:* utiliza como recurso principal formulários que são preenchidos pelos usuários, com dados relacionados ao objetivo do site.

Portal*:* Compreende um conjunto de páginas, links que convergem para um determinado contexto.

Aplicação orientada à transação*:* Neste modelo, o usuário envia solicitações que a aplicação processa em um banco de dados, retornando ao usuário a resposta de sua solicitação. (Miletto e Bertagnolli, 2014, p. 9) – em português.

(**Content site**: Displays content from a specific area linearly, including resources such as images, tables, and graphs, with simple navigation.

Data entry or user registration site*:* Uses forms as the primary feature, allowing users to submit data relevant to the website's purpose.

Portal*:* Consists of multiple pages and links that converge into a specific context.

Transaction-oriented application*:* In this model, the user sends requests that the application processes in a database, returning a response to the user's request (Miletto and Bertagnolli, 2014, p. 9) – in english).

The management system was developed using PHP programming language. This choice was made due to several advantages PHP offers, such as its speed in execution, which ensures quick responses to user requests. Its interpreted nature allows the code to run on machines without the need for robust hardware configurations, making it capable of performing efficiently even on devices with limited resources. Moreover, PHP's widespread popularity provides easy access to libraries and frameworks, facilitating development and troubleshooting. During the solution's development, the lightweight nature of PHP allowed the entire coding process to be carried out on a standard office computer with basic specifications. The essential requirements for developing the application included installing a private APACHE web server, an Integrated Development Environment (IDE) for writing the code, and a Database Management System (DBMS) specifically MySQL, with MariaDB being chosen for its lightweight and robust structure, making it ideal for working synergistically with PHP. "O PHP (um acrônimo recursivo para PHP: Hypertext Preprocessor) é uma linguagem de script open source de uso geral, muito utilizada, e especialmente adequada para o desenvolvimento web e que pode ser embutida dentro do HTML", (PHP, 2024) – em português. (PHP (a recursive acronym for PHP: Hypertext Preprocessor) is a widely used, open-source, general-purpose scripting language, especially suited for web development, and it can be embedded into HTML. (PHP, 2024)) – in english. A database system is a computerized record management system functioning as an electronic repository of data files. Users can perform various operations on these files, such as queries and updates (Date, 1975). The MariaDB organization (2024) highlights that, given this structure, the database offers a way to manipulate data: SQL.

SQL (Standard Query Language) is a powerful way to search for records or make changes to them. MariaDB was initially developed as a drop-in replacement for MySQL, with a focus on integrating application development with the PHP language. In scenarios where the transition from static pages to complex functional systems for the web is planned, Bento (2021) points out that PHP becomes the primary choice. The combination of PHP and MySQL has a long history of collaboration, being used to create web content such as news portals, online stores, blogs, and social networks. This partnership is notable for the ease with which code interacts with the database and for its simplicity and efficiency in implementing low-cost solutions. This pair is also known for the ease of writing code to interact with databases, another advantage is that PHP installations on servers and development environments generally come with built-in libraries for MySQL access by default in most cases. The author also presents several reasons why readers should adopt these technologies, summarized as follows:

- PHP nasceu para a web, e sua integração com servidores web dos mais diversos é simples. Isso facilita a manutenção de servidores e diminui as barreiras de entradas no mercado para novos produtos e serviços.
- PHP tem uma curva de aprendizado suave, comparada a outras linguagens. Isso possibilita que muitas pessoas aprendam o básico e consigam com isso testar suas ideias em projetos simples, sem muitos custos. Esses projetos podem crescer e continuar usando PHP.
- PHP abstrai muitas das rotinas mais comuns da comunicação entre clientes (navegadores) e servidores, deixando o caminho livre para que quem está desenvolvendo possa se focar na lógica do que está construindo e não em como processar uma requisição HTTP.
- PHP e MySQL são tecnologias livres, abertas e gratuitas. Mais uma vez, derrubando as barreiras dos custos de entrada no mercado para novas empresas, pessoas e projetos, por exemplo.
- É fácil encontrar serviços de hospedagem que oferecem PHP e MySQL, dos mais simples e acessíveis aos mais complexos e caros.
- Serviços de hospedagem PHP e MySQL são mais baratos que serviços semelhantes para outras tecnologias, o que facilita bastante a entrada de novas pessoas e reduz custos até para as grandes empresas.
- MySQL é leve e rápido, mesmo para quantidades razoavelmente grandes de dados. Eu já trabalhei pessoalmente por alguns anos com uma base de dados MySQL com alguns Terabytes de tamanho.
- Como muitas pessoas começam a programar para web com PHP e MySQL, é natural que elas continuem com PHP e MySQL quando evoluem na área e em suas carreiras. […] (Bento, 2021, p. 10) – em português.

- PHP was created for the web, and its integration with various web servers is simple. This facilitates server maintenance and lowers the entry barriers for new products and services.
- PHP has a smooth learning curve compared to other languages, allowing many people to quickly learn the basics and test ideas through simple projects at low costs. These projects can grow and continue using PHP.
- PHP abstracts many common routines related to communication between clients (browsers) and servers, enabling developers to focus on the logic of their applications instead of processing HTTP requests.
- PHP and MySQL are free, open, and free of cost, reducing entry barriers for new businesses, individuals, and projects.
- It is easy to find hosting services offering PHP and MySQL, ranging from the simplest and most affordable to the most complex and expensive.
- PHP and MySQL hosting are cheaper than hosting for other technologies, making it easier for new users to enter the market and reducing costs even for large companies.
- MySQL is lightweight and fast, even when dealing with reasonably large datasets. For example, the author has personally worked with a MySQL database of several terabytes over the years.
- Since many people start programming for the web with PHP and MySQL, they tend to continue using these technologies as they advance in the field and in their careers. […] (Bento, 2021, p. 10)) – in English.

The system's development followed a model designed from the analysis carried out during the requirements-gathering meetings. As illustrated in Figure 1, the planned database architecture allows for individualized general records. These records are integrated through the relationships between the database tables and the logic implemented in the code, enabling general classes, such as 'People' and 'Animals,' to assume specific roles within the system. For example, the inclusion of an attribute like CD_PESSOA_VETERINARIO_REMETENTE – em português (CD_PERSON_VETERINARIAN_SENDER – in English), which stores the code of a previously registered person, facilitates the association between the main record forms and the professionals involved.

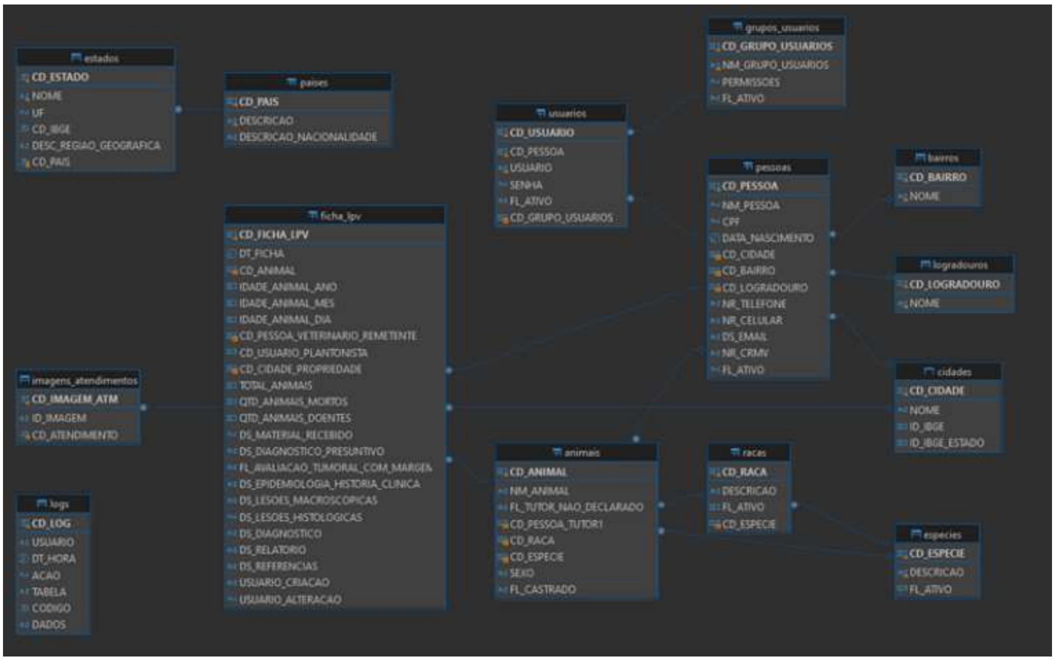


Figure 1. Final database architecture

This relational process ensures efficient data intersection, optimizing information retrieval and promoting the reuse of existing data within the system.

Object-Oriented Programming (OOP): In the development of applications using object-oriented programming (OOP), Deitel (2003) explains that it focuses on key concepts like inheritance and polymorphism. Inheritance allows for the creation of new classes based on existing ones, absorbing their attributes and behaviors while adding new, specific features. This saves time and promotes the reuse of high-quality software, reducing future problems. Polymorphism enables writing programs that work with various related classes, making it easier to add new features to the system. These techniques effectively manage the complexity of software development, allowing new classes to inherit from existing superclasses and become candidates for future subclasses.

> Objetos ou, mais precisamente, as classes de onde os objetos vêm são essencialmente componentes reutilizáveis de software. Há objetos data, objetos data/hora, objetos áudio, objetos vídeo, objetos automóveis, objetos pessoas etc. Quase qualquer substantivo pode ser razoavelmente representado como um objeto de software em termos dos atributos (por exemplo, nome, cor e tamanho) e comportamentos (por exemplo, calcular, mover e comunicar). Grupos de desenvolvimento de software podem usar uma abordagem modular de projeto e implementação orientados a objetos para que sejam muito mais produtivos do que com as técnicas anteriormente populares como "programação estruturada" [...] (Deitel, 2003, p.43) – em português.
>
> (Objects or more precisely, the classes from which objects are derived are essentially reusable software components. For example, there are objects for data, date/time, audio, video, vehicles, and people. Almost any noun can reasonably be represented as a software object, defined by its attributes (e.g., name, color, and size) and behaviors (e.g., calculating, moving, and communicating). Software development teams can use a modular approach in OOP-based design and implementation to become far more productive than with earlier techniques like "structured programming" [...] (Deitel, 2003, p. 43) – in English).

"A orientação a objetos é um paradigma que representa uma filosofia para construção de sistemas" (Dall'oglio, 2016, p.88) – em português. (Object-oriented programming is a paradigm that represents a philosophy for building systems" (Dall'oglio, 2016, p. 88) – in English).

Unlike structured languages such as Cobol, Clipper, and Pascal where systems consist exclusively of sequential procedures and variables that are not always well-organized within their context, object-oriented programming reflects a perspective more aligned with the real world (Dall'Oglio, 2016).

MVC Framework (Model, View, Controller): The system was developed using the MVC (Model-View-Controller) architecture, a widely adopted pattern in web application development due to its ability to separate the application into three main components. This approach facilitates system maintenance and scalability, enables code reuse, and supports more structured testing, which contributes to the quality and stability of the final software. The MVC architecture helps developers build applications by separating key components: data handling and storage, functions that process data input, and user interface display. MVC architecture defines where each type of logic should be in the application (Santos, Moreira, Silva, Freitas, 2010).

- **Model:** é o objeto de aplicação responsável por gerenciar a comunicação com a base de dados da aplicação, emitir erros de conexão quando necessário e sinalizar quando uma transação envolvendo manipulação de dados do usuário foi concluída com sucesso.
- **View:** é a interface visualizada pelo usuário, a região do sistema que menos contém lógica de programação, pelo fato de maior parte de sua estruturação ser voltada para construção, modelagem e renderização das páginas dinâmicas da aplicação utilizando componentes visuais como Tabelas, Gráficos, Formulários, Animações, Botões, Campos de digitação, Cores e entre outros.
- **Controller:** trabalha em relação às entradas e solicitações que o usuário inicializa a partir da View e a partir desses gatilhos, também coordena como elas reagirão para entregar a resposta da operação esperada pelo operante do software. (Gamma, Helm, Johson, Vlissides, 2000) – em português.

- **Model:** The application object responsible for managing communication with the database, issuing connection error alerts, and signaling when a transaction involving user data manipulation has been successfully completed.
- **View:** The part of the system the user interacts with. This component contains minimal programming logic, as it focuses primarily on building, modeling, and rendering the application's dynamic pages using visual elements such as tables, charts, forms, animations, buttons, input fields, and colors.
- **Controller:** Works with the inputs and requests initialized by the user from the View and coordinates how these interactions will respond to deliver the expected operation results (Gamma, Helm, Johnson, Vlissides, 2000) – in English).

Santos et al. (2010) also report that dividing the application into Model, View, and Controller components simplifies complexity management, as these parts are independent, enabling parallel development. Adding new clients becomes straightforward by simply including new views and controllers, which promotes greater code reuse. Additionally, customization and replacement of components are easier, given that the MVC pattern is highly extensible. Developers familiar with MVC benefit from easier code readability thanks to its standardized and clear structure. The combined use of object-oriented practices with the MVC design pattern enabled the effective implementation of business rules and solution workflows, leading to continuous improvements in the platform's routines, simplifying development, and enhancing the product's quality. An example of this combination can be seen in the controller for registering LPV – Laudo Patológico Veterinário – in portugês (Veterinary Pathological Report – in English) forms, illustrated in Figure 2. This controller plays a crucial role in coordinating user forms and handling user actions. Following the workflow, the controller ensures data integrity and proper handling on the server side. When all input conditions are met, the controller establishes a single connection with the database, ensuring performance while handling insert, update, and delete operations. It also interacts with other aggregated classes, such as the 'People' class, which represents the veterinarian and the pet owner, and the 'Animals' class, which represents the animal. Thus, the controller with the business rules for registering the LPV File, accesses all the classes and methods needed to be implemented within the same routine, involving data validation, security checks, database connection calls and database handling.

Implementation of application features: The software provides advanced data analysis capabilities through components such as dynamic tables and allows the export of database information in CSV format. This feature is particularly useful when urgent statistical information is needed via spreadsheet applications. The solution offers manageable access control. Like most modern systems, it supports multiple simultaneous access sessions while ensuring security by managing user permissions. This management capability gives administrators autonomy to control which users are authorized to log into the platform and access available services. Additionally, user groups enable the administrator to instantly grant or revoke access to specific parts of the system or specific screens based on the permission levels assigned to each professional. All operations performed are recorded in the database, ensuring they can be audited later if needed. Since many pathological and clinical services provided by the institution are related to research aimed at improving treatments and procedures, the application supports the collection and registration of images and digital attachments useful for these studies.

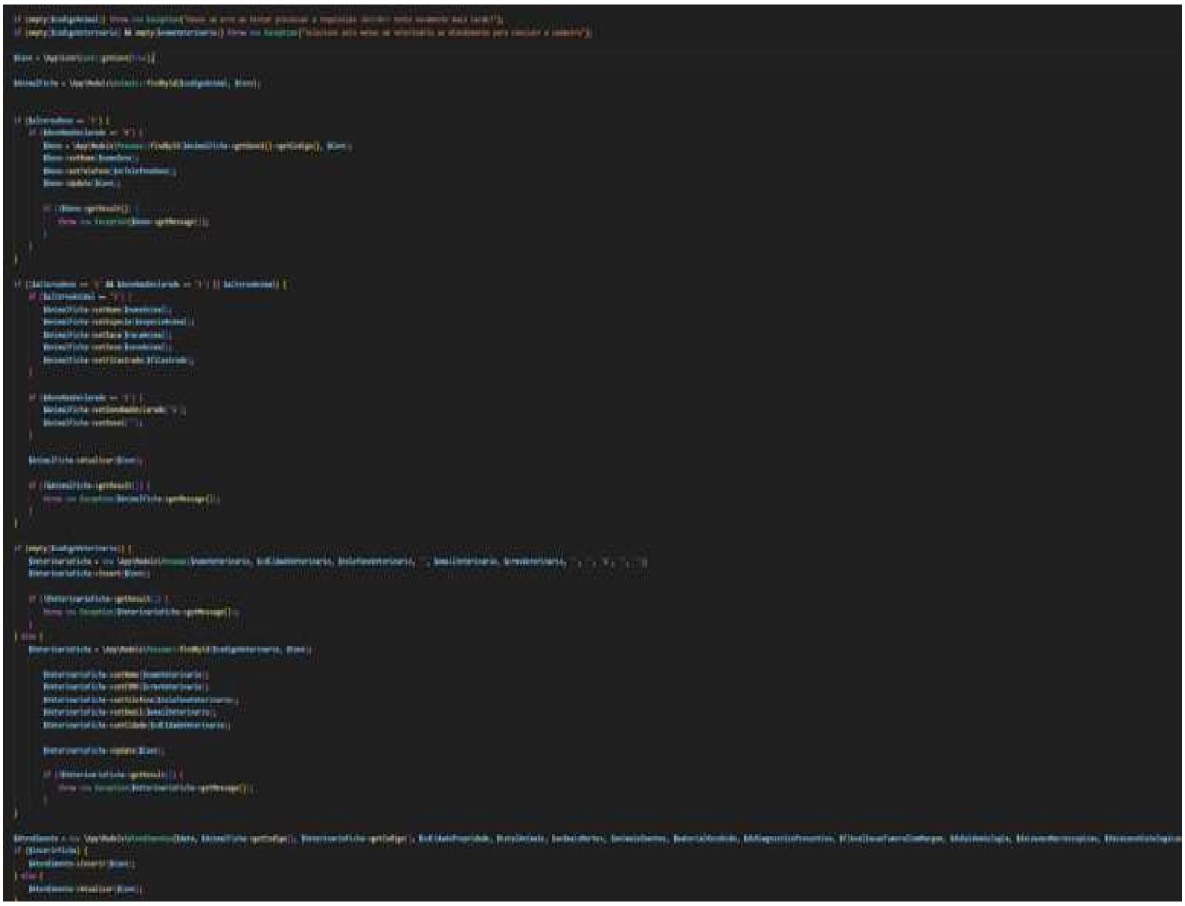


Figure 2. Code of controller with the business rules for registering the LPV file, all the classes and methods are implemented in the same routine

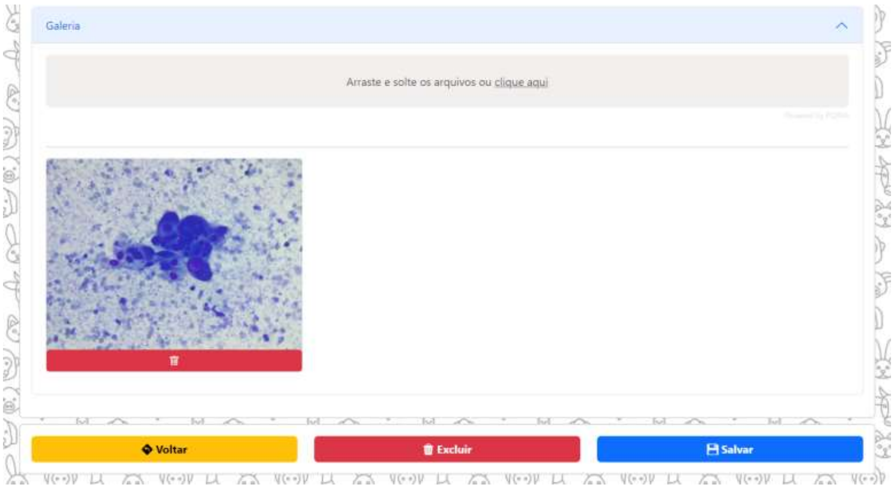


Figure 3. Pathology care image gallery panel with a microscopic analysis image attached to the record

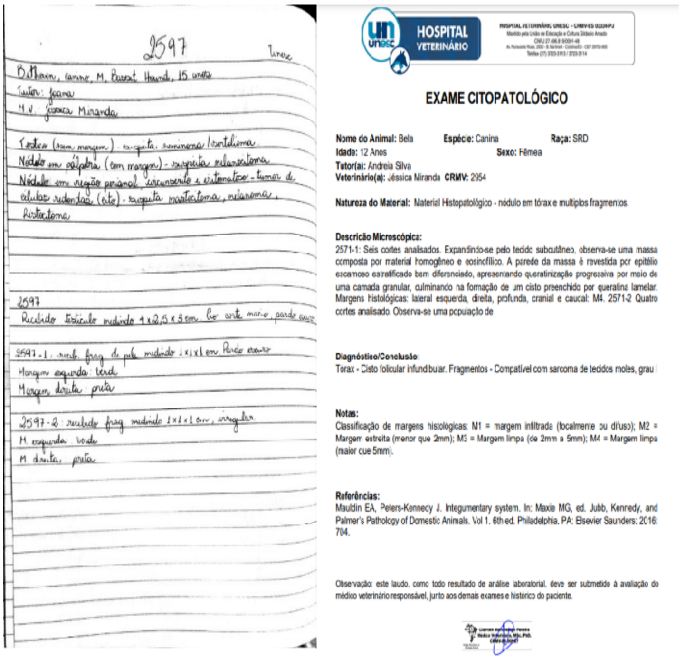
A dedicated feature was developed to centralize these attachments in the case files. When creating or updating a case file, the user can import these files directly into the system, where they are stored centrally in the cloud server. As shown in figure 3, when the user accesses the gallery module when registering the pathological service, they can attach documents or files that are important for the studies, such as a recorded image of a microscopic analysis. The generation of diagnostic and clinical service reports is a key functionality of the solution. Once a case file is completed with all the necessary information, the print functionality becomes available, allowing users to choose from three main document templates: Cytopathological Examination, Histopathological Examination, and Examination Results. The system offers downloads in both PDF format, ensuring a standardized and unmodifiable document, and DOCX format, which can be edited through office applications while maintaining the institution's layout standards.

The application's responsiveness was carefully designed to ensure usability on the two main platforms currently in demand: desktop and mobile. This significantly enhances the solution's accessibility and effectiveness, making it practical and efficient for users on various devices. To maximize the system's reach and accessibility, the final solution was hosted on a cloud server. The choice of hosting service and domain was carefully made, considering the system's specific features and requirements.

The selected cloud server not only provided the necessary environment for system operation but also offered additional resources that added value to the project. These resources included native HTTPS security and an SSL certificate to ensure the protection of transmitted data, daily backups of files and databases to maintain data integrity, and disaster recovery capabilities to minimize the risk of data loss in emergencies.

RESULTS

After the development and hosting of the proposed solution, the results indicate a significant improvement in the quality of veterinary pathological study procedures. The implementation of the system brought meaningful changes to the workflow and information security within the sector. Previously, the pathological data registration process faced several obstacles that compromised efficiency. One of the main issues was that only one professional could perform registrations at a time, creating a bottleneck in the workflow. Additionally, physical documents presented readability inconsistencies, varying according to the handwriting of the professional filling them out, which complicated future use of the data. The risks associated with physical records, such as fire, exposure to liquids, or the natural degradation of documents posed a constant threat to information integrity. With the implementation of the developed solution, these problems were eliminated. Records are now digital and centralized, with online access available for both mobile devices and computers, enabling multiple professionals to work simultaneously on different records in a standardized way, following designated processes. Data entry and document generation now take only a few minutes, optimizing working time. Figure 4 illustrates these improvements. On the left-hand side, it shows the old physical records, often with erasures and corrections made with a pen, as well as being limited to the physical format. In contrast, the right-hand side shows a report issued from the system, in PDF format, of the Cytopathological Examination model. In this report, the registered information is organized in a standardized way and presented in an official document of the institution, highlighting the uniformity and clarity achieved with the new platform.



2597 Unesc

Bethoven, canino, M, Basset Hound, 15 anos
Tutor: Joana
M.V.: Jéssica Miranda

Tórax (sem margem) - suspeita: [illegible]
Nódulo em pálpebra (com margem) - suspeita: melanocitoma
Nódulo em região perianal, circunscrito e eritematoso - tumor de células redondas (alto) - suspeita: mastocitoma, melanoma, histiocitoma

2597
Recebido testículo medindo 4x2,5x3 cm. Ao corte macio, pardo claro

2597-1: receb. frag. de pele medindo 1x1x1 cm. Pardo escuro
Margem esquerda: verde
Margem direita: preta

2597-2: recebido frag. medindo 1x1x1 cm, irregular
M. esquerda: verde
M. direita: preta

HOSPITAL VETERINÁRIO UNESC - CMRV-ES 0220-PJ
Mantido pela União de Educação e Cultura Sildásio Amado
CNPJ 27.068.8 9/0001-48
Av. Fernando Rosa, 200 - B. Vila Itapema - Colatina/ES - CEP 29700-000
Telefax (27) 3723-3113 / 3723-3114

EXAME CITOPATOLÓGICO

Nome do Animal: Bela **Espécie:** Canina **Raça:** SRD
Idade: 12 Anos **Sexo:** Fêmea
Tutor(a): Andreia Silva
Veterinário(a): Jéssica Miranda **CRMV:** 2954

Natureza do Material: Material Histopatológico - nódulo em tórax e multiplos fragmentos.

Descrição Microscópica:
2571-1: Seis cortes analisados. Expandindo-se pelo tecido subcutâneo, observa-se uma massa composta por material homogêneo e eosinofílico. A parede da massa é revestida por epitélio escamoso estratificado bem diferenciado, apresentando queratinização progressiva por meio de uma camada granular, culminando na formação de um cisto preenchido por queratina lamelar. Margens Histológicas: lateral esquerda, direita, profunda, cranial e caudal: M4. 2571-2 Quatro cortes analisado. Observa-se uma população de

Diagnóstico/Conclusão:
Tórax - Cisto folicular infundibular. Fragmentos - Compatível com sarcoma de tecidos moles, grau I

Notas:
Classificação de margens histológicas: M1 = margem infiltrada (focalmente ou difuso); M2 = Margem estreita (menor que 2mm); M3 = Margem limpa (de 2mm a 5mm); M4 = Margem limpa (maior que 5mm).

Referências:
Mauldin EA, Peters-Kennedy J. Integumentary system. In: Maxie MG, ed. Jubb, Kennedy, and Palmer's Pathology of Domestic Animals. Vol 1. 6th ed. Philadelphia, PA: Elsevier Saunders; 2016: 704.

Observação: este laudo, como todo resultado de análise laboratorial, deve ser submetido à avaliação do médico veterinário responsável, junto aos demais exames e histórico do paciente.

Figure 4. Document of a pathology study produced manually, on the left-hand side of the image. On the right, a cytopathological examination report issued by the system developed.

DISCUSSION

At the conclusion of this research, the application of object-oriented programming (OOP) revealed notable differences in approaches and challenges as discussed by authors such as Deitel (2003) and Dall'Oglio (2007). Deitel (2003) emphasizes OOP within a Java-centric context, asserting its essential role while highlighting the steep learning curve associated with mastering these techniques, particularly for novice developers. This complexity can pose significant barriers to adopting OOP, often making its implementation challenging and, in certain scenarios, impractical. Conversely, Dall'Oglio (2007) offers a more accessible and pragmatic perspective on OOP, especially with the evolution of scripting languages like PHP. According to him, PHP has evolved over the years to incorporate object-oriented principles in a simpler and more flexible manner, offering a streamlined syntax that eliminates much of the complexity. This evolution allows for practical achievement of advanced prototype results or even a final version of management software, as developed in this research. The transition from a physical workflow to a fully digital environment led to notable improvements in the quality and security of processed information. The developed solution addressed previous issues such as inconsistencies in records and limited simultaneous access, thereby enhancing productivity and increasing the reliability of stored data. Furthermore, object-oriented architecture provided significant gains in code organization and maintainability, facilitating scalability during production. The integration of suitable technologies, grounded in a solid theoretical and methodological foundation, demonstrated the potential for substantial improvements in internal processes of institutions, such as the veterinary hospital examined in this study. The collaboration with industry professionals allowed for the creation of a system that effectively met their needs, ensuring a more efficient and secure workflow. This research serves as a practical example of the importance of aligning theory with practice in the field of technology, emphasizing the value of customized and high-quality solutions in diverse areas involving data management.

REFERENCES

Bento, E. J. 2021. Desenvolvimento web com PHP e MySQL. Casa do Código. Disponível em: https://books.google.com.br/books?id=xG2CCwAAQBAJ&printsec=frontcover&hl=pt-BR&source=gbs_atb#v=onepage&q&f=false

Boccato, V. R. C. (2006). Metodologia da pesquisa bibliográfica na área odontológica e o artigo científico como forma de comunicação. Revista de Odontologia da Universidade de São Paulo. v.18, n.3. https://arquivos.cruzeirodosuleducacional.edu.br/principal/old/revista_odontologia/pdf/setembro_dezembro_2006/metodologia_pesquisa_bibliografica.pdf

Dall'Oglio, P. (2016). PHP Programando com Orientação a Objetos. Novatec Editora Ltda. https://www.adianti.com.br/phpoo_mostra.pdf

Date, C. J. (2004). Introdução a sistemas de bancos de dados. Elsevier Editora LTDA. 8ª ed. https://www.google.com.br/books/edition/Introdu%C3%A7%C3%A3o_a_sistemas_de_bancos_de_dados/xBeO9LSlK7UC?hl=pt-BR&kptab=editions&gbpv=1

Deitel, P., & Deitel, H. (2003). Java como programar (C. A. L. Lisbôa, Trad.). Porto Alegre: Bookman.

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (2000). Padrões de projeto: soluções reutilizáveis de software orientado a objetos. Porto Alegre: Bookman.

MariaDB. (n.d.). Introdução a banco de dados relacionais. https://mariadb.com/kb/en/introduction-to-relational-databases

Miletto, E., & Bertagnolli, S. (2014). Desenvolvimento de Software II: Introdução ao desenvolvimento web com HTML, CSS, Javascript e PHP. Porto Alegre: Bookman. https://books.google.com.br/books?hl=pt-BR&lr=&id=lcLFAwAAQBAJ&oi=fnd&pg=PR1&dq=artigo+sobre+desenvolvimento+web&ots=kSLHpA83Cx&sig=IcZ1AGsWOo2_K4SBe4xRd5Aq65w#v=onepage&q&f=false

PHP. (n.d.). Introdução ao PHP. https://www.php.net/manual/pt_BR/intro-whatis.php

Santos, I., Gomes, M., & Oliveira, P. (2010). Possibilidades e limitações da arquitetura MVC (Model-View-Controller) com ferramenta IDE (Integrated Development Environment). Revista de Estudos em Engenharia e Computação, 3(2). https://revistas.unifenas.br/index.php/RE3C/article/view/65/24

Thiollent, M. (2009). Metodologia de pesquisa-ação. São Paulo: Cortez Editora. https://marcosfabionuva.com/wp-content/uploads/2018/08/7-metodologia-da-pesquisa-ac3a7c3a3o.pdf

****