DESENVOLVIMENTO WEB: SISTEMA DE CONTROLE PARA ENTRADA E SAÍDA DE VISITANTES

Guilherme Freitas Paixão, Leonardo Kenzo Okuno, Pedro Henrique Macedo da Silva, Wagner dos Santos Clementino de Jesus.

Fundação Vale Paraibana de Ensino/ Colégio Técnico Antônio Teixeira Fernandes Unidade Centro, R. Paraibuna, 75 - Jardim São Dimas, São José dos Campos - SP, 12245-020, univapguilhermefreitaspaixao@gmail.com, leonardo.okuno999@gmail, pedrohenriquemacedo1910@gmail.com, wagner@univap.br.

Resumo

O presente trabalho propõe o desenvolvimento de uma aplicação web para controle de entrada e saída de visitantes em ambientes corporativos, garantindo armazenamento seguro, rastreabilidade e confiabilidade dos registros. Foram utilizadas ferramentas como Visual Studio Code e MySQL Workbench, com programação em JavaScript e back-end em Node.js e Express. O sistema segue a arquitetura API (Application Programming Interface) REST (Representational State Transfer) e o padrão MVC (Model-View-Controller), permitindo comunicação eficiente cliente-servidor e separação de responsabilidades. A segurança das informações foi assegurada por autenticação JWT (JSON Web Token), prevenindo acessos não autorizados, enquanto o banco de dados relacional garante integridade e consistência dos dados. A aplicação demonstra eficiência no gerenciamento de visitantes, funcionários e veículos, integrando segurança física e lógica, contribuindo para a continuidade operacional, mitigação de riscos e fortalecimento da governança corporativa.

Palavras-chave: Controle de entrada e saída. Segurança da informação. Aplicação web. Banco de dados. JWT.

Curso: Técnico em Informática.

Introdução

A segurança da informação é um dos pilares para proteção de dados e processos em ambientes corporativos (SCHNEIER, 2015). Com a digitalização, garantir integridade, disponibilidade e confidencialidade tornou-se essencial, exigindo políticas e protocolos eficientes para prevenir acessos não autorizados e assegurar a continuidade organizacional.

Segurança física e lógica são complementares: a primeira protege instalações e ativos tangíveis, enquanto a segunda protege dados e sistemas por meio de autenticação multifator, controle de acesso, criptografia e monitoramento contínuo (STALLINGS, 2018).

Sistemas de controle de acesso são estratégicos na gestão de visitantes e funcionários, utilizando biometria, cartões magnéticos, leitores digitais e softwares integrados a bancos de dados, garantindo rastreabilidade e monitoramento (HENNESSY; PATTERSON, 2013).

A automação e integração de sistemas reduzem falhas humanas e fortalecem a governança de segurança. Este trabalho propõe o desenvolvimento de uma aplicação web para controle de entrada e saída de visitantes, integrando bancos de dados robustos e mecanismos de segurança que asseguram confidencialidade, integridade e disponibilidade das informações (DATE, 2004; LAUDON; LAUDON, 2016).

Metodologia

A pesquisa foi conduzida a partir do levantamento de requisitos junto aos usuários do sistema, por

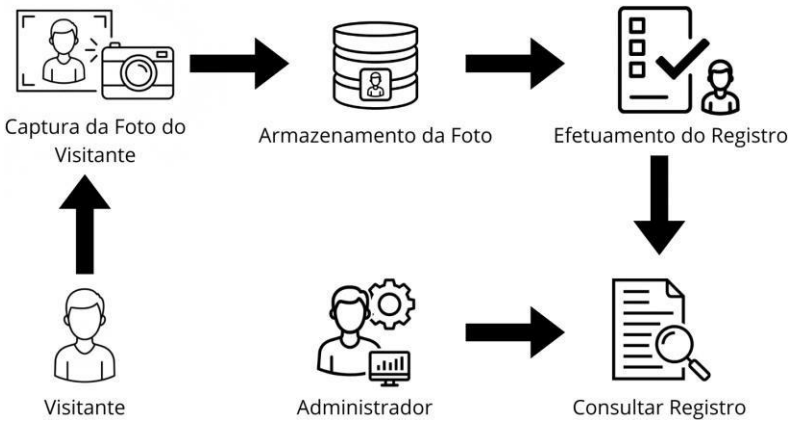
meio de reuniões e questionários, passando por validação em demonstração prática pelos usuários do sistema do Parque Tecnológico Univap. Além de utilizar conceitos de sistemas precedentes como referência para melhorias. No que se refere à segurança, diversas práticas foram implementadas para garantir a integridade e a confiabilidade do sistema. O acesso à aplicação foi protegido por autenticação JWT (JSON Web Token), que assegura a transmissão segura de informações entre cliente e servidor, com tokens gerados no login e verificados a cada requisição (RICHARDSON; RUBY, 2007; STAMMER, 2020; FERNANDES; OLIVEIRA, 2019). Os tokens possuem tempo de expiração de 24 horas, prevenindo sessões indevidas prolongadas. Além disso, foram utilizados prepared statements no banco de dados, mitigando ataques de SQL Injection, e o framework bcrypt foi aplicado para hashing de senhas, aumentando a resistência contra acessos não autorizados. Para reduzir riscos de inconsistências ou dados maliciosos, foi implementada também a validação de entradas tanto no front-end quanto no back-end. Por meio do padrão MVC (Model-View-Controller), separando responsabilidades entre Model, Controller e View (FOWLER, 2002; GAMMA et al., 1994), auxiliando no desenvolvimento por seções isoladas, trazendo maior rapidez e eficiência no desenvolvimento.

Para edição e programação foi utilizado o Visual Studio Code, compatível com linguagens como JavaScript, Python, C#, Java, HTML e CSS (Cascading Style Sheets) (MICROSOFT, 2021), enquanto o MySQL Workbench foi empregado para criação e gerenciamento do banco de dados, permitindo modelagem visual, execução de consultas SQL (Structured Query Language) e administração de usuários (DUCHARME, 2018).

A linguagem escolhida foi JavaScript, devido à sua simplicidade e compatibilidade com navegadores modernos, possibilitando sistemas responsivos e interativos (FLANAGAN, 2020). No back-end, utilizou-se Node.js (Node JavaScript Runtime), ambiente assíncrono eficiente em operações de I/O (Input/Output), e o framework Express, que facilita a criação de APIs (Application Programming Interfaces) e gerenciamento de rotas, garantindo robustez e manutenção simplificada (HOLMES, 2017; TILFORD, 2019). O sistema segue a arquitetura API (Application Programming Interface) REST (Representational State Transfer), utilizando métodos HTTP (Hypertext Transfer Protocol) como GET, POST, PUT e DELETE para manipulação de recursos (FIELDING, 2000; PAUTASSO; ZIMMERMANN; LEYMANN, 2008).

A Figura 1 mostra um diagrama em blocos representando a sequência de funcionamento da aplicação desenvolvida.

Figura 1 – Sequência de funcionando do sistema



Fonte: Autores, 2025

Resultados

Os resultados demonstraram que o sistema desenvolvido atendeu aos requisitos de gerenciamento de visitantes em ambientes corporativos, garantindo o armazenamento seguro das informações. O cadastro ocorreu de forma eficiente, registrando nome, RG e fotografia, vinculados ao banco de dados MySQL com hash para proteção das imagens.

Nos testes realizados, o tempo médio para cadastrar um visitante foi de 5 segundos, enquanto a consulta por RG retornou resultados em menos de 5 segundos também. Comparado ao processo manual usado pelos usuários do sistema, que não possui maneira significativa em agilidade, houve um ganho inestimável em eficiência e velocidade. Além do ganho em velocidade, esta aplicação implementa consultas de dados, que possibilita aos usuários, maior confiabilidade e segurança nas informações do visitante. A Figura 2 mostra a interface de captura de fotos pelos funcionários

Figura 2 – Tela de Cadastro

Fonte: Autores, 2025

Discussão

O estudo demonstra a eficiência de uma aplicação web para controle de entrada e saída de visitantes para o Parque Tecnológico Univap. O sistema permite cadastrar, consultar, alterar, excluir visitantes e todos dados pertinentes à entidade visitante. Em comparação a sistemas anteriormente usados, esta aplicação se destaca e diferencia pelo ganho na segurança e velocidade, além de apresentar funcionalidades inéditas com a versatilidade de uma aplicação web, podendo ser acessada por diferentes computadores sem a necessidade de instalação de arquivos independentes.

Futuras melhorias são, mecanismos de notificação em tempo real, como alertas automáticos por e-mail e SMS que podem ser disparados em situações de tentativas de acesso não autorizado. Outra melhoria seria o desenvolvimento de relatórios gerenciais mais robustos, permitindo análises detalhadas sobre os fluxos de visitantes, incluindo identificação de horários de pico, estatísticas de frequência por período e indicadores de reincidência.

Conclusão

Os resultados deste estudo confirmam que o sistema web desenvolvido para controle de entrada e saída de visitantes, utilizando Node.js, padrão MVC e armazenamento seguro de fotos via hash, atingiu os objetivos propostos, oferecendo agilidade, segurança e confiabilidade na gestão de registros. O sistema mostrou eficiência no gerenciamento de visitantes e permite melhorias futuras em relatórios e interfaces. Assim, o trabalho contribui para a modernização e automação do controle de visitantes, integrando programação, segurança da informação e banco de dados, tornando-se uma aplicação que agiliza e otimiza o funcionamento para o fluxo de entrada e saída na instituição Parque tecnológico Univap.

Referências

ALMEIDA, Fernanda. *Segurança da informação em sistemas corporativos: estratégias e boas práticas*. São Paulo: Editora TechBooks, 2021.

DATE, C. J. *Introdução a sistemas de banco de dados*. 8. ed. Rio de Janeiro: Elsevier, 2004.

DUCHARME, John. *MySQL Cookbook: Solutions for Database Developers and Administrators*. 2. ed. Sebastopol: O'Reilly Media, 2018.

FERNANDES, João Paulo; OLIVEIRA, Ricardo. *Autenticação JWT com Node.js: Como proteger APIs REST com tokens seguros*. São Paulo: Novatec, 2019.

FIELDING, Roy Thomas. *Architectural Styles and the Design of Network-based Software Architectures*. Tese (Doutorado em Ciência da Computação) – University of California, Irvine, 2000.

FLANAGAN, David. *JavaScript: The Definitive Guide*. 7. ed. Sebastopol: O'Reilly Media, 2020.

FOWLER, Martin. *Patterns of Enterprise Application Architecture*. Boston: Addison-Wesley, 2002.

GAMMA, Erich et al. *Design Patterns: Elements of Reusable Object-Oriented Software*. Boston: Addison-Wesley, 1994.

HENNESSY, John L.; PATTERSON, David A. *Arquitetura de computadores: uma abordagem quantitativa*. 5. ed. São Paulo: Elsevier, 2013.

HOLMES, Ethan. *Express in Action: Writing, building, and testing Node.js applications*. Shelter Island: Manning, 2017.

LAUDON, K. C.; LAUDON, J. P. *Sistemas de informação gerenciais*. 14. ed. São Paulo: Pearson, 2016.

MICROSOFT. *Visual Studio Code Documentation*. Microsoft, 2021. Disponível em: https://code.visualstudio.com/docs.

MORAES, Ricardo. *Aplicações web e sistemas de controle de acesso em instituições públicas e privadas*. Rio de Janeiro: Editora Ciência e Tecnologia, 2019.

PAUTASSO, Cesare; ZIMMERMANN, Olaf; LEYMANN, Frank. *RESTful web services vs. "big" web services: making the right architectural decision*. In: Proceedings of the 17th International Conference on World Wide Web. New York: ACM, 2008. p. 805–814.

SCHNEIER, B. *Data and Goliath: the hidden battles to collect your data and control your world*. New York: W.W. Norton & Company, 2015.

STALLINGS, W. *Cryptography and network security: principles and practice*. 7. ed. Boston: Pearson, 2018.

STAMMER, Cody. *Node.js Web Development: Server-side development with Node 14 using frameworks such as Express, Docker, MongoDB, and AWS*. 5. ed. Birmingham: Packt Publishing, 2020.

TILFORD, Sam. *Node.js in Action: Server-side JavaScript for Web Development*. 2. ed. Shelter Island: Manning, 2019