
SISTEMA DE GERENCIAMENTO DE GUARDAS (TIRO DE GUERRA)

GUARD MANAGEMENT SYSTEM (MILITARY TRAINING UNIT)

Aleff Silva

Tecnólogo em Análise e Desenvolvimento de Sistemas

Centro Universitário Einstein de Limeira- UniEinstein

aaleffsilvaa@gmail.com

Gabriel Madeira do Carmo

Tecnólogo em Análise e Desenvolvimento de Sistemas

Centro Universitário Einstein de Limeira- UniEinstein

gabrielmadeira26@gmail.com

João Pedro Furlan

Tecnólogo em Análise e Desenvolvimento de Sistemas

Centro Universitário Einstein de Limeira- UniEinstein

joaopedrofurlan12@gmail.com

Thiago Salhab Alves

Mestre em Ciência da Computação

Centro Universitário Einstein de Limeira- UniEinstein

thiagosalhab@gmail.com

RESUMO

O processo de organização da escala de guarda no Tiro de Guerra apresentou-se suscetível a falhas em razão do controle manual, evidenciando a necessidade de modernização. Este artigo teve como objetivo desenvolver um sistema para automatizar a geração da escala de guarda dos atiradores. Empregou-se o modelo incremental de desenvolvimento de software descrito por Pressman e Maxim (2016, p. 43-44), que permite a construção do sistema em pequenos ciclos funcionais, facilitando ajustes e validações contínuas. Utilizou-se a linguagem Python (Django) para o back-end e Nuxt/Vue para o front-end. O sistema desenvolvido possibilitou a organização automática das escalas conforme critérios pré-estabelecidos, visando substituir o antigo método manual baseado em planilhas. Embora a implementação ainda dependa da

aceitação pelo Tiro de Guerra, os resultados indicaram que a solução pode proporcionar maior precisão, agilidade e mitigação de falhas no gerenciamento das escalas. Concluiu-se que a adoção do sistema tende a otimizar o processo e a contribuir para uma gestão mais eficiente e segura.

Palavras-chave: Tiro de Guerra. Sistema de Escala. Automatização. Desenvolvimento de Software. Gestão de Escalas.

ABSTRACT

The process of organizing the guard duty roster at the military training unit proved to be susceptible to errors due to manual control, highlighting the need for modernization. This study aimed to develop a system to automate the generation of guard duty schedules for the trainees. The incremental software development model described by Roger Pressman (2016) was employed, allowing the system to be built in small functional cycles, facilitating continuous adjustments and validations. Python (Django) was used for the back-end, while Nuxt/Vue was adopted for the front-end. The developed system enabled the automatic organization of schedules according to predefined criteria, aiming to replace the previous manual method based on spreadsheets. Although the implementation still depends on approval by the military training unit, the results indicated that the solution may provide greater accuracy, agility, and mitigation of failures in schedule management. It was concluded that adopting the system tends to optimize the process and contribute to more efficient and secure management.

Keywords: Military Training Unit. Duty Roster System. Automation. Software Development. Schedule Management.

INTRODUÇÃO

A organização da escala de guarda no âmbito do Tiro de Guerra é atualmente realizada de forma manual, o que demanda um esforço considerável tanto do sargento responsável quanto dos atiradores envolvidos. O processo consiste na divulgação semanal da escala em grupos de mensagens instantâneas, como o WhatsApp, e, em caso de necessidade de troca de guardas, exige que os atiradores negociem entre si, comuniquem pessoalmente o sargento e aguardem a atualização manual da escala. Cada atualização requer a exclusão da escala anterior e o envio de uma nova versão, o que torna o gerenciamento mais complexo e suscetível a atrasos e desencontros de informações.

Diante desse cenário, torna-se evidente a importância da modernização dos processos internos por meio da automação. A utilização de sistemas informatizados permite não apenas reduzir o retrabalho, como também promove maior organização, agilidade e confiabilidade nas operações cotidianas (Pressman, 2016). A automação de tarefas administrativas que antes dependiam de planilhas ou de controles informais em aplicativos de mensagens vem se consolidando como uma prática essencial para garantir a eficiência e a segurança dos dados em diferentes instituições.

O uso de sistemas web em ambientes militares tem se mostrado relevante diante da necessidade de gestão eficaz de informações, comunicação entre os membros da organização e atualização contínua dos dados. Segundo Lindsay (2020), os sistemas digitais passaram a mediar diversas atividades relacionadas à coleta, ao armazenamento, à apresentação, à análise e à comunicação de informações nas organizações militares, tornando as práticas de informação um componente importante para a efetividade das atividades institucionais.

Diante do exposto, este artigo teve como objetivo desenvolver um sistema web para automatizar a organização da escala de guarda do Tiro de Guerra, de modo a otimizar a comunicação e reduzir o retrabalho associado à metodologia manual atualmente utilizada.

DESENVOLVIMENTO

A pesquisa foi conduzida com base na análise do processo de organização da escala de guarda do Tiro de Guerra, sem a necessidade de interação direta com os atiradores. O foco do estudo foi o desenvolvimento de um sistema para otimizar e automatizar as tarefas realizadas

pelo sargento responsável pela gestão da escala. Para isso, o estudo se fundamentou na observação das práticas adotadas no Tiro de Guerra e nas informações coletadas junto ao sargento, que permitiram compreender as necessidades e os desafios do processo atual de escalas.

Tecnologias utilizadas

A escolha das tecnologias empregadas no desenvolvimento considerou as características do sistema proposto, especialmente a necessidade de automatização de processos, organização das informações, comunicação entre diferentes componentes da aplicação e facilidade de manutenção. Nesse sentido, a utilização de uma arquitetura baseada em tecnologias web possibilita centralizar as funcionalidades do sistema e disponibilizá-las aos usuários por meio de uma interface acessível por navegador. Para Pressman e Maxim (2016), a engenharia de software deve considerar processos sistemáticos e técnicas que contribuam para a construção de produtos confiáveis e passíveis de manutenção, aspecto particularmente importante em sistemas destinados ao gerenciamento de informações institucionais.

A linguagem Python foi utilizada na implementação do back-end da aplicação. Segundo Borges (2014), Python é uma linguagem de programação de alto nível, multiparadigma e que dispõe de uma ampla biblioteca padrão, características que favorecem sua utilização em diferentes tipos de aplicações, inclusive aplicações web. No contexto deste projeto, a linguagem foi empregada para implementar as regras de negócio responsáveis pelo gerenciamento dos usuários, escalas, trocas de guardas e demais operações necessárias ao funcionamento do sistema. A escolha também está relacionada à possibilidade de organizar o código de maneira modular, facilitando sua evolução durante os ciclos de desenvolvimento.

Para a construção da camada de back-end foi utilizado o framework Django. Frameworks de desenvolvimento de software permitem estruturar a aplicação a partir de componentes e práticas previamente estabelecidos, reduzindo a necessidade de implementar repetidamente funcionalidades comuns. Greenfeld e Greenfeld (2017) destacam, no contexto do desenvolvimento com Django, a importância de boas práticas relacionadas à organização do projeto, segurança, modelos, formulários e desenvolvimento de aplicações web. Dessa maneira, o Django foi empregado como elemento estruturante da aplicação, concentrando a implementação das regras de negócio e a comunicação com o banco de dados.

Na camada de apresentação, foram utilizados JavaScript, Vue.js e Nuxt.js. O Vue.js é um framework progressivo destinado à construção de interfaces de usuário e permite estruturar aplicações por meio de componentes reutilizáveis (VUE.JS, 2026). A adoção de uma abordagem baseada em componentes favorece a organização da interface e possibilita separar visualmente as diferentes funcionalidades do sistema, como autenticação, calendário, cadastro de usuários e solicitação de trocas. O Nuxt.js, por sua vez, amplia o ecossistema Vue.js e fornece recursos para a construção de aplicações web, incluindo mecanismos de roteamento e diferentes estratégias de renderização (Kok, 2020). No projeto desenvolvido, essas tecnologias foram utilizadas para proporcionar uma interface organizada e facilitar a interação dos usuários com as funcionalidades disponibilizadas.

O armazenamento das informações foi realizado por meio do MySQL, utilizando o MySQL Workbench como ferramenta de apoio à estruturação e gerenciamento do banco de dados. Para Elmasri e Navathe (2018), sistemas de banco de dados fornecem mecanismos para organizar, armazenar e recuperar informações de forma estruturada, sendo fundamentais para aplicações que necessitam manter dados persistentes e relacionamentos entre diferentes entidades. No sistema desenvolvido, o banco de dados é responsável pelo armazenamento das informações relacionadas aos usuários, escalas, guardas e solicitações de troca, permitindo que esses dados sejam posteriormente consultados e manipulados pelas demais camadas da aplicação.

A comunicação entre as partes da aplicação também foi verificada por meio de requisições HTTP, utilizando-se o Insomnia durante o processo de testes. Essa etapa permitiu verificar o comportamento das rotas disponibilizadas pelo back-end e observar se os dados enviados pelo front-end eram corretamente processados e retornados. A utilização de uma ferramenta específica para inspeção das requisições contribuiu para identificar inconsistências durante o desenvolvimento e apoiar a validação das funcionalidades implementadas.

A comparação com a literatura reforça o impacto positivo da informatização de processos administrativos. Conforme aponta Pressman (2016), sistemas bem projetados reduzem falhas, promovem a eficiência e garantem maior confiabilidade dos dados.

Verificação, validação e qualidade do software

A avaliação do sistema também foi orientada por referências normativas relacionadas à documentação de testes, verificação e validação e qualidade de produtos de software. Nesse

contexto, considerou-se a IEEE 829, norma tradicionalmente associada à documentação de testes de software e sistemas. A IEEE 829-2008 estabelece elementos para a documentação das atividades de teste, contribuindo para registrar de maneira estruturada o planejamento, os procedimentos, os resultados e as evidências obtidas durante a avaliação do software (IEEE, 2008). Embora essa norma tenha sido posteriormente substituída pela série ISO/IEC/IEEE 29119, sua estrutura permanece relevante como referência histórica para organização da documentação de testes.

Também foi considerada a IEEE 1012, relacionada aos processos de verificação e validação (V&V). A verificação busca avaliar se os produtos resultantes de uma determinada atividade de desenvolvimento estão de acordo com os requisitos estabelecidos, enquanto a validação procura verificar se o produto atende ao uso pretendido e às necessidades dos usuários (IEEE, 2024). No contexto do sistema desenvolvido, esses princípios foram considerados durante a análise das funcionalidades, das regras de geração das escalas, das solicitações de troca e do comportamento das diferentes permissões de acesso.

Para complementar a avaliação, adotou-se como referência o modelo de qualidade definido pela ISO/IEC 25010. A norma estabelece um modelo de qualidade de produtos de software e sistemas, permitindo analisar diferentes características e subcaracterísticas de qualidade ao longo do ciclo de vida do produto (ISO, 2023). Para o sistema de gerenciamento de guardas, aspectos como adequação funcional, usabilidade, confiabilidade, segurança, eficiência de desempenho e manutenibilidade são particularmente relevantes, uma vez que a aplicação precisa produzir escalas de acordo com regras definidas, apresentar informações de maneira compreensível e manter os dados protegidos e disponíveis aos usuários autorizados.

A consideração conjunta dessas referências proporciona maior rigor à avaliação do sistema, pois permite relacionar os testes realizados às práticas de documentação, verificação, validação e qualidade de software. Dessa forma, a análise não se restringe à constatação de que as funcionalidades foram executadas, mas busca estabelecer critérios mais sistemáticos para avaliar se o sistema atende aos requisitos definidos e se apresenta características de qualidade compatíveis com sua finalidade.

A Figura 1 ilustra a tela inicial do sistema que permite o acesso ao sistema por meio de autenticação com usuário e senha. Verifica o tipo de usuário (atirador ou administrador) e redireciona para o painel adequado conforme as permissões.

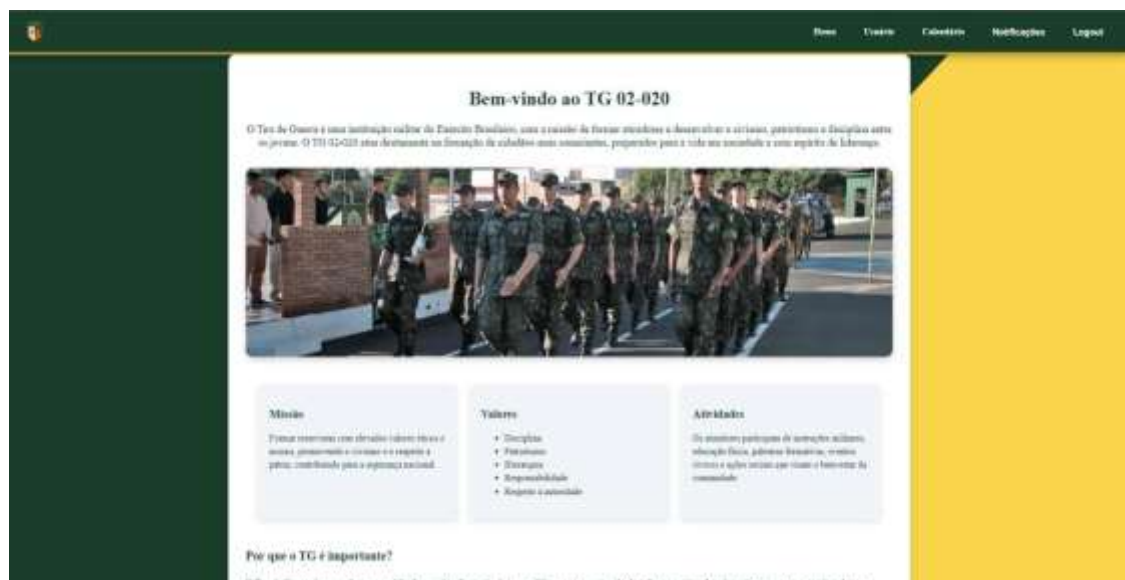
Figura 1 – Tela de login



Fonte: os autores

A Figura 2 apresenta informações institucionais sobre o Tiro de Guerra e acesso rápido às seções do sistema via barra de navegação (Home, Perfil, Trocas, Calendário). Pode incluir mensagens de boas-vindas ou comunicados.

Figura 2 – Tela Home (Pós-Login)



Fonte: os autores

A Figura 3 disponível apenas para administradores do sistema, permite cadastrar novos usuários, possui um formulário intuitivo com campos validados e botão de envio e mostra todos os usuários já cadastrados, permitindo também editar ou excluir usuários existentes.

Figura 3 – Tela de cadastro de usuário



Nome	Email	CPF	Ações
gabriel	gabriel@gmail.com	123	Editar Excluir
marcelo	marcelo@gmail.com	1234	Editar Excluir
joão	joao@gmail.com	12345	Editar Excluir
ana	ana@gmail.com	1234	Editar Excluir
marcelo	marcelo@gmail.com	12345	Editar Excluir
marcelo	marcelo@gmail.com	1234	Editar Excluir

Fonte: os autores

A Figura 4 mostra a escala de guardas com visual mensal usando FullCalendar. Atiradores apenas veem suas escalas; administradores podem sortear ou apagar escalas.

Figura 4 – Tela de Calendário

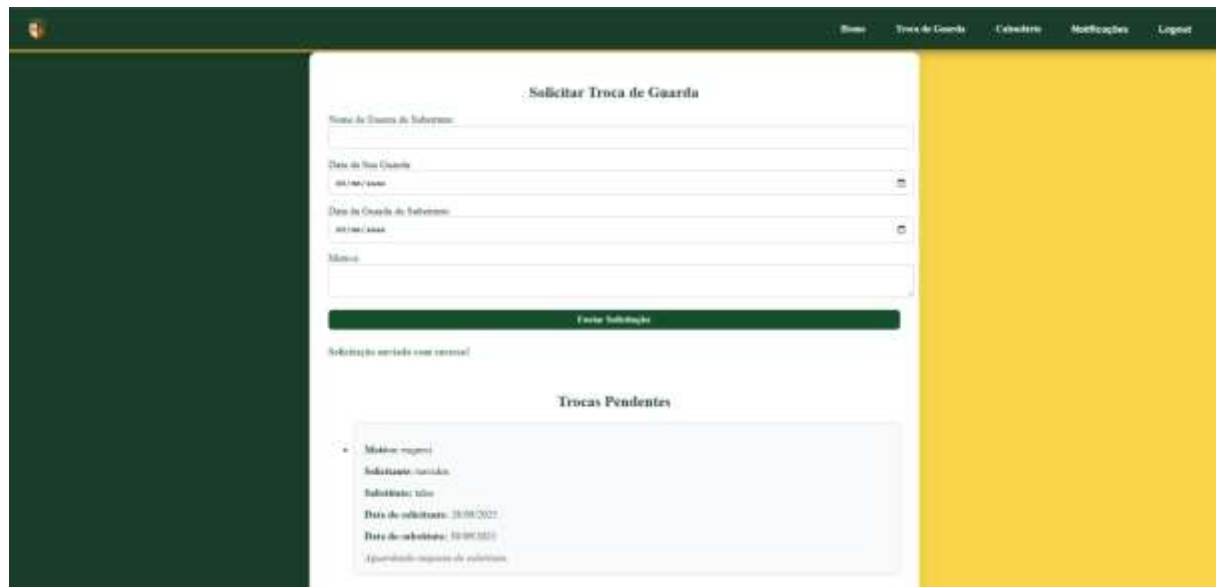


Nome	Seg	Ter	Qua	Qui	Sex	Sáb
Guarda 1	Guarda 2	Guarda 3	Guarda 4	Guarda 5	Guarda 6	Guarda 7
Guarda 8	Guarda 9	Guarda 10	Guarda 11	Guarda 12	Guarda 13	Guarda 14
Guarda 15	Guarda 16	Guarda 17	Guarda 18	Guarda 19	Guarda 20	Guarda 21
Guarda 22	Guarda 23	Guarda 24	Guarda 25	Guarda 26	Guarda 27	Guarda 28
Guarda 29	Guarda 30	Guarda 31	Guarda 32	Guarda 33	Guarda 34	Guarda 35

Fonte: os autores

A Figura 5 permite solicitação de troca de guarda entre atiradores. Exibe formulário e gerencia as trocas ainda pendentes.

Figura 5 - Tela de Troca de Guarda

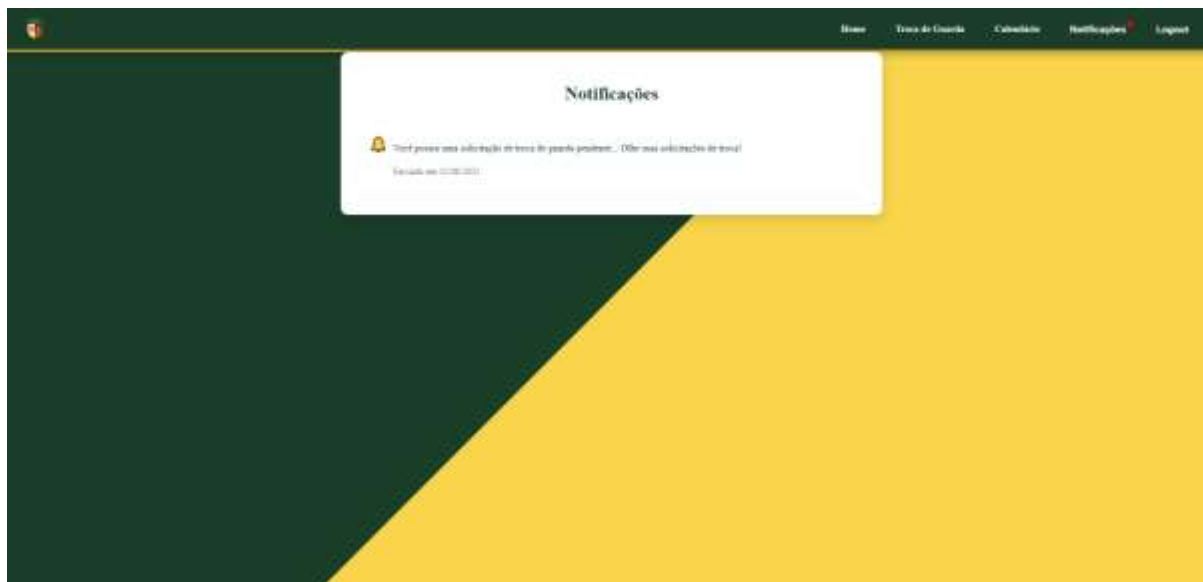


The screenshot displays a web application interface for requesting a guard swap. The top navigation bar includes links for 'Home', 'Troca de Guarda', 'Calendário', 'Notificações', and 'Logout'. The main content area is titled 'Solicitar Troca de Guarda' and contains a form with the following fields: 'Nome do Guarda do Substituto', 'Data da Sua Guarda' (with a calendar icon), 'Data da Guarda do Substituto' (with a calendar icon), and 'Motivo'. A green 'Enviar Solicitação' button is positioned below the form. A confirmation message 'Solicitação enviada com sucesso!' is displayed. Below this, a section titled 'Trocas Pendentes' shows a list of pending requests, including details like 'Motivo: viagem', 'Solicitante: Ricardo', 'Substituto: João', 'Data de solicitação: 28/04/2022', 'Data de substituição: 02/05/2022', and 'Aguardando resposta do solicitante'.

Fonte: os autores

A Figura 6 informa sobre as trocas solicitadas entre usuários. O sargento, também vê essa tela com todas as trocas que foram feitas, para melhor controle.

Figura 6 - Sistema de Notificações



Fonte: os autores

CONSIDERAÇÕES FINAIS

O desenvolvimento do Sistema de Gerenciamento de Guardas mostrou-se uma solução viável e eficaz para modernizar a gestão das escalas no Tiro de Guerra. Com base nas simulações realizadas, foi possível verificar ganhos em organização, precisão e redução de retrabalho.

A utilização de tecnologias modernas como Django e Nuxt.js permitiu a construção de uma aplicação web robusta, com bom desempenho e fácil manutenção. A validação com usuários reais evidenciou que o sistema pode facilitar a rotina do sargento responsável e melhorar a comunicação entre os atiradores.

Como sugestões para trabalhos futuros, recomenda-se: A integração com sistemas de autenticação institucional; A implementação de notificações automáticas por e-mail ou WhatsApp; A criação de um módulo de relatórios e estatísticas para análise de desempenho dos atiradores.

A adoção do sistema pode representar um avanço na digitalização dos processos internos e servir como modelo para outras unidades semelhantes.

REFERÊNCIAS

BORGES, L. E. *Python para desenvolvedores: aborda Python 3.3*. São Paulo: Novatec, 2014. 320 p.

ELMASRI, R.; NAVATHE, S. B. *Sistemas de banco de dados*. 7. ed. São Paulo: Pearson, 2018.

GREENFELD, D. R.; GREENFELD, A. R. *Two Scoops of Django: Best Practices for Django 1.11*. 5. ed. [S.l.]: Two Scoops Press, 2017.

IEEE. *IEEE Standard for Software and System Test Documentation*. IEEE Std 829-2008. New York: Institute of Electrical and Electronics Engineers, 2008.

IEEE. *IEEE Standard for System, Software, and Hardware Verification and Validation*. IEEE Std 1012-2024. New York: Institute of Electrical and Electronics Engineers, 2024.

ISO. *ISO/IEC 25010:2023: Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — Product quality model*. Geneva: International Organization for Standardization, 2023.

KOK, L. T. *Hands-on Nuxt.js Web Development: Build universal and static-generated Vue.js applications using Nuxt.js*. Birmingham: Packt Publishing, 2020.

PRESSMAN, R. S.; MAXIM, B. R. *Engenharia de software: uma abordagem profissional*. 8. ed. Porto Alegre: AMGH, 2016.

LINDSAY, J. R. *Information Technology and Military Power*. Ithaca: Cornell University Press, 2020.

SILVA, J. L. *et al. Sistemas Web: conceitos e práticas*. 2. ed. São Paulo: Atlas, 2017.

VUE.JS. *Vue.js: The Progressive JavaScript Framework*. 2026. Disponível em: documentação oficial do Vue.js. Acesso em: 7 ago. 2026.