

Organização: Projeto-2-VA-ES

Membros: Luiz Fellipe | Victor Winicius | Rian Wilker | Luan Valentino

Postmortem

Resumo do que funcionou e do que não funcionou durante a iteração, e planos para melhorar o que não funcionou

a) Período: **21 de março de 2023 - 04 de abril de 2023**

b) O que estava planejado?

- atividades e para quem:

4º e última iteração:

- Seis refatorações; (Rian, Luan e Luiz)
- Navegação adequada das Views; (Rian e Luiz)
- Diagrama de classes UML; (Victor)
- Testes executados corretamente no Actions; (Luan, Rian, Luiz e Victor)
- Projeto implantado no Railway; (Luan e Luiz)
- Postmortem da iteração; (Luan)
- Release publicado. (Victor)

c) O que foi feito?

- atividades e por quem:

4º iteração:

- Seis refatorações; (Rian, Luan e Luiz)
- Navegação adequada das Views; (Rian e Luiz)
- Diagrama de classes UML; (Victor)
- Testes executados corretamente no Actions; (Luan, Rian, Luiz e Victor)
- Projeto implantado no Railway; (Luan e Luiz)
- Postmortem da iteração; (Luan)
- Release publicado. (Victor)

d) O que não foi feito?

- atividades e por quem, quais os impedimentos:

Todas as iterações foram feitas por completo e divididas para cada integrante do grupo.

e) O que está planejado para a próxima iteração?

- atividades e para quem:

Não haverá próxima iteração.

f) Lições aprendidas

1. A usabilidade das Views é crucial para a experiência do usuário. É importante dedicar tempo para testar e ajustar a navegação, garantindo que o usuário possa encontrar facilmente as informações necessárias e realizar as ações desejadas.
2. A refatoração contínua do código é fundamental para manter a qualidade do software e facilitar a manutenção futura. É importante dedicar tempo para entender o código existente e identificar oportunidades de melhoria, mesmo que isso possa exigir algum esforço extra no curto prazo.
3. Um diagrama de classes UML pode ser uma ferramenta valiosa para visualizar e comunicar a estrutura do software. No entanto, é importante lembrar que o diagrama deve refletir o código atual e ser atualizado conforme o software evolui. Além disso, é preciso ter cuidado para não se prender demais ao diagrama e perder de vista o objetivo final do software.