

RELATÓRIO FINAL DO PROJETO DA DISCIPLINA TESTE DE SOFTWARE

Título: BookSwap - Sistema de gerenciamento de troca de livros

Componentes: Dayvison Eryc de Moura Silva, Claudio Caueh Oliveira Xavier, Jhoan Fernandes de Oliveira.

1. Introdução:

O “BookSwap” é um aplicativo que muda a maneira como os leitores interagem com seus livros e uns com os outros, ele permite que os usuários cadastrem seus próprios livros e os troquem com outros, criando uma comunidade de leitores engajados, este aplicativo não só incentiva a leitura, mas também promove a sustentabilidade ao reutilizar livros e proporciona uma maneira econômica de explorar novos títulos e autores.

- a. Justificativa:** O aplicativo é importante por várias razões. Primeiramente, ele promove a leitura e a educação, permitindo que os usuários tenham acesso a uma variedade de livros que talvez não pudessem adquirir de outra forma. BookSwap também ajuda a construir comunidades de leitores, permitindo que pessoas com interesses literários semelhantes se conectem e compartilhem suas paixões. E por fim, o aplicativo é uma solução econômica para os leitores. Em vez de comprar novos livros, os usuários podem trocar os livros que já leram por outros que desejam ler, economizando dinheiro e recursos.
- b. Objetivo geral:** O objetivo principal do nosso aplicativo é criar uma plataforma de troca de livros que seja fácil de usar, econômica e sustentável. Pretendemos alcançar uma comunidade de leitores engajados que possam compartilhar e descobrir novos livros de uma maneira que seja benéfica para todos os usuários.

2. Metodologia:

Inicialmente o diagrama de classes foi feito para ter-se uma noção de como o programa funcionaria. Nisto, chegamos à conclusão de que três entidades, ou objetos do mundo real, seriam o coração do projeto: livro, biblioteca e usuário. Por conseguinte, o código foi desenvolvido na linguagem de programação Java com base nos campos e

métodos contidos no diagrama de classes. Além disso, utilizamos de testes unitários com o auxílio do JUnit5 e realizamos testes de integração e de sistema. O diagrama, conforme o código era escrito, foi passando por pequenas alterações. Dessa forma, basta apenas que o computador tenha instalado as ferramentas necessárias para executar uma aplicação Java. Por fim, os padrões MVC e Observer foram usados.

3. Resultados:

Após a utilização do programa, este apresentou resultados satisfatórios para um projeto que deve ser melhor aprimorado pelos desenvolvedores. Inicialmente o programa mostra ao usuário uma tela de login e cadastro. Caso o usuário escolha fazer o cadastro, uma nova “janela” será aberta perguntando alguns dados sobre o usuário, como, por exemplo, o email e senha. Ao fazer o login, uma nova tela é exibida: esta demonstra as funcionalidades da biblioteca. Quanto à principal função do BookSwap, a de troca de livros, o usuário logado na biblioteca primeiro solicita o *id* do livro que ele deseja e em seguida o *id* do livro que tem e deseja trocar. Quando a solicitação é feita, uma mensagem é enviada para a caixa de mensagem do destinatário. Este usuário, por sua vez, consegue visualizar as solicitações de trocas pendentes e, assim, confirmar a troca.

4. Qualidade de Software:

O programa BookSwap foi desenvolvido seguindo o padrão de arquitetura de software Model-View-Controller (MVC). Este padrão foi escolhido por sua eficácia em separar a lógica de negócios (Model), a interface do usuário (View) e o controle do fluxo de dados (Controller), facilitando a manutenção e a compreensão do código. Além disso, adotamos uma abordagem de **refatoração contínua** para melhorar a qualidade do código e a manutenção do sistema. Isso envolveu revisar e aprimorar o código regularmente, garantindo que ele seguisse as melhores práticas e padrões de design.

Ademais, implementamos um processo de **revisão por pares** rigoroso, onde cada alteração de código era meticulosamente examinada por outro membro da equipe antes de ser integrada ao repositório principal. Esse processo não só ajudou a detectar e corrigir erros precocemente, mas também promoveu a colaboração e o compartilhamento de conhecimento entre os membros da equipe.

5. Conclusão:

A realização deste trabalho apresentou vários desafios e dificuldades. Um dos

principais deles foi garantir que a maioria dos cenários possíveis fossem cobertos pelos testes. Isso exigiu uma compreensão profunda das regras de negócio e dos requisitos do sistema.

Além disso, a correção dos erros identificados durante os testes também foi um desafio. Em muitos casos, a solução exigiu uma mudança na lógica do código e uma compreensão clara de como diferentes partes do código interagem entre si. Outro desafio encontrado foi aplicar os inúmeros conceitos da orientação a objetos, assim como fazer o diagrama de classes.

Apesar desses desafios, a realização deste trabalho foi uma oportunidade valiosa para aprender e aplicar práticas de desenvolvimento de software de alta qualidade. Através da aplicação do padrão MVC e da realização de testes rigorosos, conseguimos desenvolver um software robusto e confiável que atende às necessidades dos usuários.

6. Link para Repositório com demais documentações do projeto e relatório completo de testes:

< <https://github.com/JhoanDev/BookSwap> >.