

Nome: João Pedro Felicio Prudencio

Tópicos:

- 1- Descrição do Problema: Sistema de Livraria**
- 2- Requisitos do software**
- 3- Estimativa de duração do projeto completo**
- 4- Diagrama de Classes do Projeto UML**

1- Descrição do Problema: Sistema de Livraria

Escopo do Software:

O código implementa um sistema simples de livraria com funcionalidades destinadas a dois tipos de usuários: Administrador e Cliente. O software mantém uma lista de livros, cada um representado pela classe Livro. Cada livro possui atributos como título, autor, gênero, quantidade disponível e quantidade comprada.

O sistema é composto pelas seguintes classes:

- Livro (Classe de Entidade):

Representa um livro com atributos como título, autor, gênero, quantidade disponível e quantidade comprada.

- Sistema (Classe de Controle):

Responsável por armazenar a lista total de livros disponíveis no sistema.

- UsuarioAdministrador (Classe de Controle):

Oferece funcionalidades específicas para o usuário com perfil de administrador.

Métodos:

inserir_novo_livro: Adiciona um novo livro à lista total, verificando se um livro com o mesmo título já existe.

visualizar_estoque: Exibe o estoque atual, incluindo título, autor, gênero e quantidade disponível.

aumentar_quantidade_estoque: Aumenta a quantidade disponível de um livro no estoque.

- UsuarioCliente (Classe de Controle):

Oferece funcionalidades específicas para o usuário com perfil de cliente.

Métodos:

comprar_livro: Permite ao cliente comprar um livro, diminuindo a quantidade disponível e incrementando a quantidade comprada.

listar_livros_comprados: Lista os livros que foram comprados pelo cliente, incluindo título e quantidade comprada.

ver_dados_livro: Exibe os dados de um livro específico, incluindo título, autor e gênero.

visualizar_estoque: Permite ao cliente visualizar o estoque atual.

Stakeholders (Partes Interessadas):

- Administrador:

Interessado em gerenciar o catálogo de livros, adicionando novos livros, visualizando o estoque e aumentando a quantidade disponível de determinados livros.

- Cliente:

Interessado em comprar livros, listar os livros comprados, visualizar dados de um livro específico e verificar o estoque disponível.

- Resumo do Funcionamento:

Inicialização:

O sistema é inicializado com uma lista vazia de livros.

Menu Principal:

Os usuários têm a opção de entrar como Administrador, Cliente ou sair do programa.

Administração (Administrador):

O Administrador pode inserir novos livros, visualizar o estoque e aumentar a quantidade disponível de livros existentes.

Cliente (UsuarioCliente):

O Cliente pode comprar livros, listar os livros comprados, ver os dados de um livro específico e visualizar o estoque.

Observações:

O sistema utiliza prints para fornecer feedback ao usuário, indicando a realização de operações.

A detecção de erros ou condições especiais é tratada por meio de mensagens impressas no console.

2- Requisitos do software

Requisitos Funcionais:

1- Inserir Novo Livro (Usuário Administrador):

O sistema deve permitir que o usuário administrador insira um novo livro com suas informações, incluindo título, autor, gênero e quantidade disponível em estoque.

2- Visualizar Estoque (Usuário Administrador):

O sistema deve permitir que o usuário administrador visualize o estoque atual, exibindo todos os livros cadastrados, seus detalhes e a quantidade disponível.

3- Aumentar Quantidade em Estoque (Usuário Administrador):

O sistema deve permitir que o usuário administrador incremente a quantidade disponível em estoque de um livro específico.

4- Comprar Livro (Usuário Cliente):

O sistema deve permitir que o usuário cliente compre um livro, reduzindo a quantidade disponível em estoque e incrementando a quantidade comprada.

5- Listar Livros Comprados (Usuário Cliente):

O sistema deve permitir que o usuário cliente visualize a lista de livros que foram comprados, exibindo seus títulos e a quantidade comprada.

6- Ver Dados do Livro (Usuário Cliente):

O sistema deve permitir que o usuário cliente visualize os detalhes (gênero, autor) de um livro específico.

7- Visualizar Estoque (Usuário Cliente):

O sistema deve permitir que o usuário cliente visualize o estoque atual, exibindo todos os livros cadastrados, seus detalhes e a quantidade disponível.

Requisitos Não Funcionais:

1- Usabilidade:

A interface do usuário deve ser intuitiva e amigável para facilitar o uso pelos administradores e clientes.

2- Desempenho:

O sistema deve ser responsivo, com tempos de resposta rápidos, garantindo uma experiência eficiente.

Histórias de Usuário:

1- Como usuário administrador, eu quero poder inserir informações de um novo livro para manter o catálogo atualizado.

2- Como usuário administrador, eu quero visualizar o estoque atual para tomar decisões informadas sobre a gestão de livros.

3- Como usuário administrador, eu quero poder aumentar a quantidade disponível em estoque de um livro específico para gerenciar a oferta.

4- Como usuário cliente, eu quero comprar um livro para adquirir uma cópia do meu interesse.

5- Como usuário cliente, eu quero listar os livros que comprei para lembrar das minhas aquisições.

6- Como usuário cliente, eu quero ver os detalhes de um livro para obter mais informações sobre o conteúdo.

7- Como usuário cliente, eu quero visualizar o estoque atual para saber quais livros estão disponíveis.

Protótipos do Sistema:

1- Inserir Novo Livro:

Protótipo de tela/formulário para inserir informações de um novo livro.

2- Visualizar Estoque:

Protótipo de tela que exibe o catálogo de livros e suas quantidades disponíveis para o usuário administrador.

3- Aumentar Quantidade em Estoque:

Protótipo de tela/formulário para selecionar um livro e incrementar sua quantidade em estoque.

4- Comprar Livro:

Protótipo de tela/formulário para selecionar e comprar um livro.

5- Listar Livros Comprados:

Protótipo de tela que exibe a lista de livros comprados.

6- Ver Dados do Livro:

Protótipo de tela que mostra os detalhes de um livro específico.

7- Visualizar Estoque (Cliente):

Protótipo de tela que exibe o catálogo de livros e suas quantidades disponíveis para clientes.

3- Estimativa de duração do projeto completo

Passo 1- Classificar como Entrada Externa (EE), Saída Externa (SE) ou Consulta Externa (CE)

Funcionalidades:

Inserir Novo Livro (Usuário Administrador): (EE)

Visualizar Estoque (Usuário Administrador): (CE)

Aumentar Quantidade em Estoque (Usuário Administrador): (EE)

Comprar Livro (Usuário Cliente): (EE)

Listar Livros Comprados (Usuário Cliente): (CE)

Ver Dados do Livro (Usuário Cliente): (CE)

Mensagens emitidas pelo sistema:

Mensagem de Inserção de Novo Livro (Usuário Administrador): (SE)

Mensagem de Visualização de Estoque (Usuário Administrador): (SE)

Mensagem de Aumento de Quantidade de Livro no Estoque (Usuário Administrador): (SE)

Mensagem de Compra de Livro (Usuário Cliente): (SE)

Mensagem de Listagem de Livros Comprados (Usuário Cliente): (SE)

Mensagem de Ver Dados do Livro (Usuário Cliente): (SE)

Mensagem do Menu Principal: (SE)

Mensagem do Menu Administrador: (SE)

Mensagem do Menu Cliente: (SE)

Mensagem de Confirmação de Venda (Usuário Cliente): (SE)

Passo 2- Cálculo do PFNA (considerando complexidade média)

$$EE = 3 \times 4 = 12$$

$$SE = 10 \times 5 = 50$$

$$CE = 3 \times 4 = 12$$

$$PFNA = 74$$

Passo 3- Campos “Respostas” da fórmula $PF = PFNA \times (0,65 + 0,01 \times \Sigma \text{Respostas})$

Necessita de backup? Nota: 5

Necessita de mecanismos especializados de comunicação? Nota: 3

Tem processamento distribuído? Nota: 2

Precisa de alto desempenho? Nota: 4

Terá grande número de usuários em paralelo? Nota: 3

Precisará de entrada de dados on-line? Nota: 4

No caso de entradas on-line, existirão múltiplas telas? Nota: 4

A atualização das entidades será feita on-line? Nota: 3

As entradas e saídas de dados serão complexas? Nota: 2

O processamento interno será complexo? Nota: 3

O código será projetado para ser reutilizado? Nota: 3

Migração e instalação estarão incluídos? Nota: 3

O sistema será instalado em diversas organizações? Nota: 2

O projeto pretende facilitar mudanças e operação do usuário? Nota: 5

Resultados= 46

Passo 4- Cálculo do $PF = PFNA \times (0,65 + 0,01 \times \Sigma \text{Respostas})$

$$PF = 74 \times (0.65 + 0.01 \times 46)$$

$$PF = 82.14$$

Passo 5- Cálculo do Esforço e da duração com COCOMO.

O peso utilizado para conversão de PFNA em LOC será o 53 da linguagem Java, pois não foi possível encontrar o valor da linguagem Python. Usando os valores de um Projetos simples: fácil entendimento e equipe pequena.

$$PFNA = 74$$

$$74 \times 53 = 3922 \text{ LOCs} = 3,922 \text{ KLOCs}$$

$$\text{Esforço} = (3,922)^{(1,05)} \times 2,4 = 10.07$$

$$\text{Duração} = (10.07)^{(0,38)} \times 2,5 = 6.01$$

4- Diagrama de Classes do Projeto UML:

