BookWorms®

SOFTWARE DEVELOPMENT

Sobre:

Na BookWorms, acreditamos no poder transformador da leitura e do conhecimento. Fundada por um grupo de apaixonados por tecnologia e literatura, nossa missão é conectar pessoas ao universo dos livros por meio de soluções inovadoras e acessíveis.

Combinando expertise em desenvolvimento de software com um profundo respeito pelo papel das bibliotecas na sociedade, criamos o BW Library, um sistema de empréstimos de livros projetado para simplificar a gestão e melhorar a experiência dos leitores.

Nosso compromisso é empoderar bibliotecas a alcançarem mais leitores, oferecendo ferramentas tecnológicas que promovem a eficiência, a inclusão e a cultura. Com cada projeto, buscamos reforçar nosso propósito: unir tecnologia e conhecimento para um futuro melhor.

Venha fazer parte dessa jornada com a gente!

BookWorms
SOFTWARE DEVELOPMENT

Funcionários :

Gabriel Mazilão Ferreira da Silva - Testador
João Pedro Dutra Coutinho - Desenvolvedor
João Victor Alves Campos - Product Owner
Luiz Eduardo Ferreira Netto - Analista de Sistemas
Ruan de Toledo Barros Lamarca - Desenvolvedor
Victor Hugo Meurer Ribeiro - Tech Lead

BookWorms
SOFTWARE DEVELOPMENT

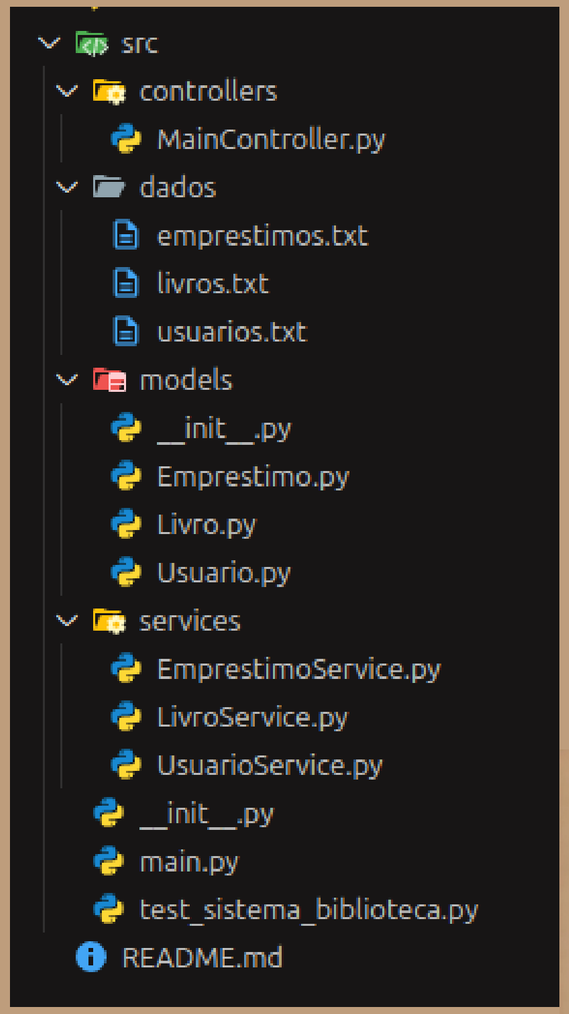
BW Library:

O BW Library é um sistema de gerenciamento de biblioteca desenvolvido com foco em facilitar o empréstimo e a administração de livros. O programa é estruturado seguindo os princípios da programação orientada a objetos e a arquitetura em camadas (model, service, controller), o que garante maior organização e facilidade de manutenção.

BookWorms
SOFTWARE DEVELOPMENT

Estrutura do Sistema:

- Model: Define as classes principais como Usuário, Livro e Empréstimo.
- Service: Realiza operações de negócio, como carregar dados e gerenciar regras.
- Controller: Gerencia a interação com o usuário e controla o fluxo do programa.

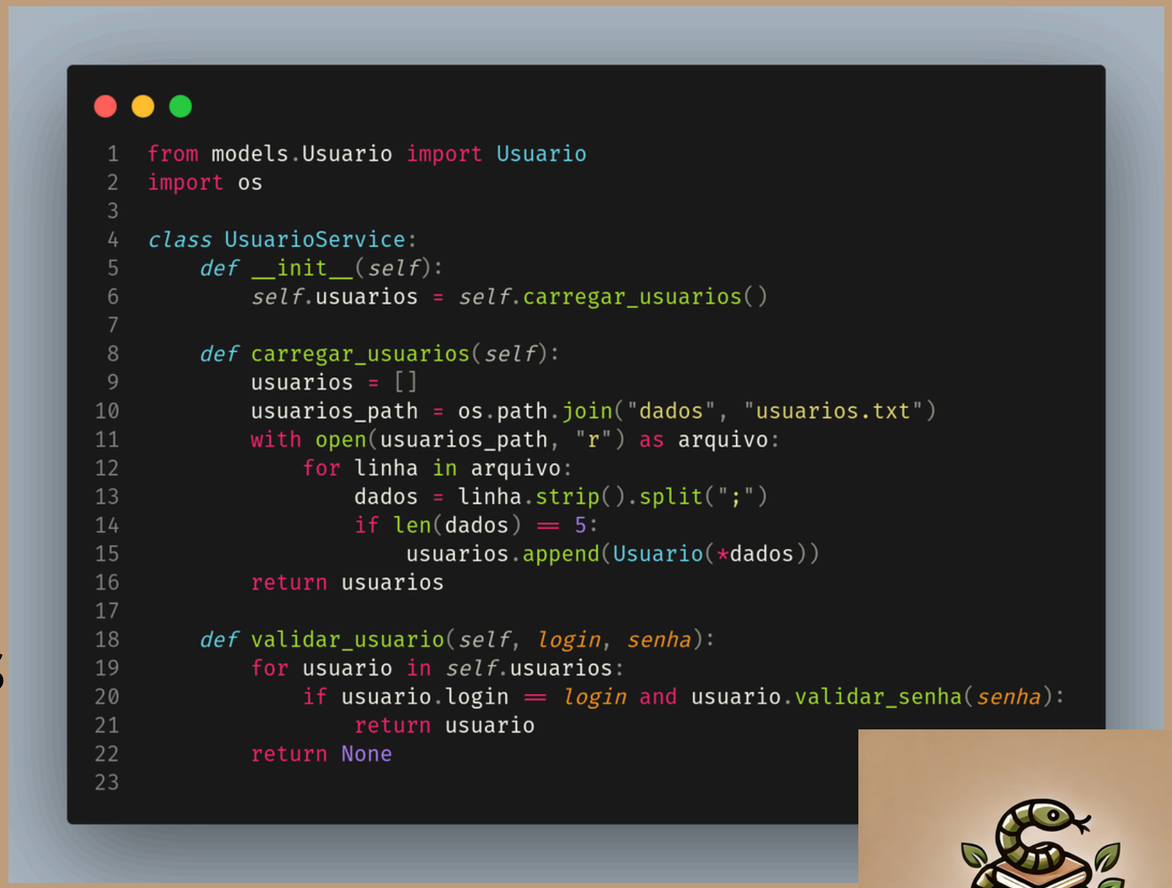


BookWorms
SOFTWARE DEVELOPMENT

Principais Funcionalidades:

Gestão de Usuários:

- Permite autenticação segura com login e senha.
- Carrega informações dos usuários a partir de um arquivo de dados.



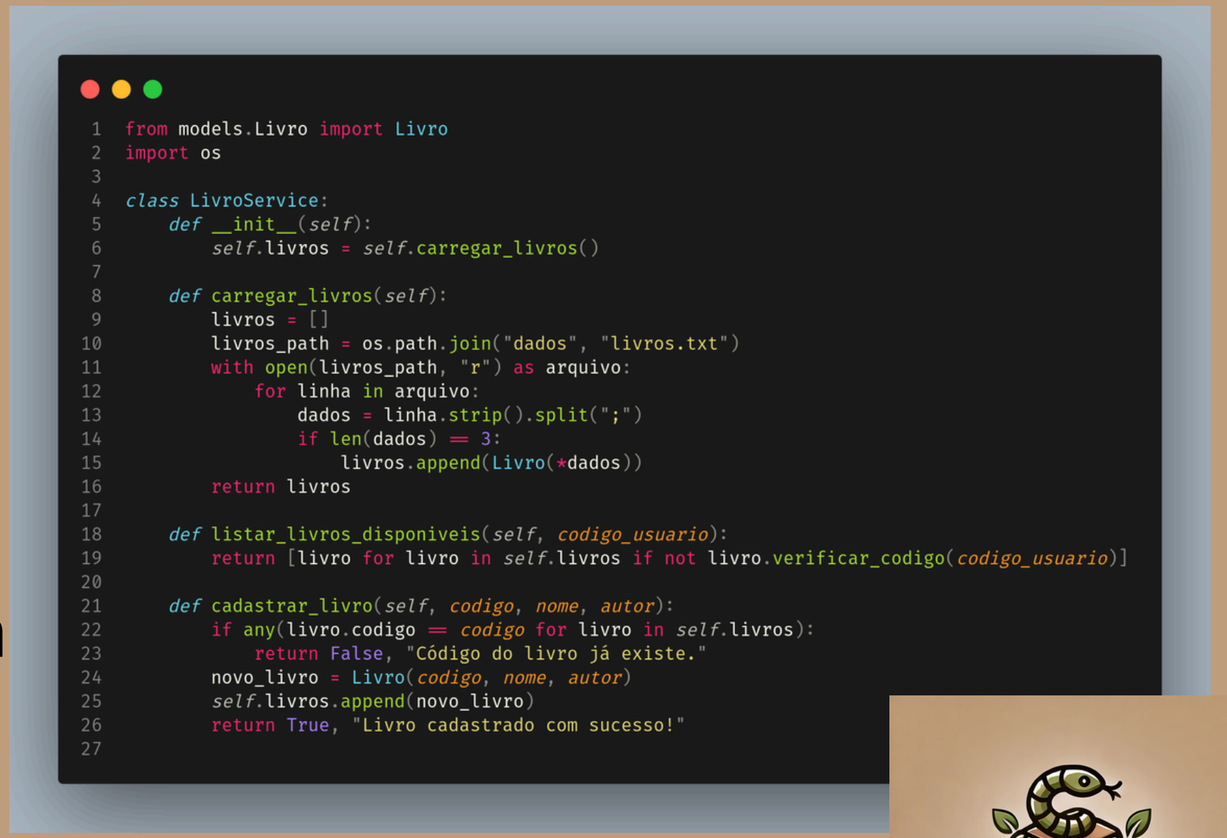
```python
from models.Usuario import Usuario
import os

class UsuarioService:
    def __init__(self):
        self.usuarios = self.carregar_usuarios()

    def carregar_usuarios(self):
        usuarios = []
        usuarios_path = os.path.join("dados", "usuarios.txt")
        with open(usuarios_path, "r") as arquivo:
            for linha in arquivo:
                dados = linha.strip().split(";")
                if len(dados) == 5:
                    usuarios.append(Usuario(*dados))
        return usuarios

    def validar_usuario(self, login, senha):
        for usuario in self.usuarios:
            if usuario.login == login and usuario.validar_senha(senha):
                return usuario
        return None

```

BookWorms
SOFTWARE DEVELOPMENT

Principais Funcionalidades:

Gerenciamento de Livros:

- Cadastro e listagem de livros disponíveis na biblioteca.
- Controle de disponibilidade com base nos empréstimos ativos.



```python
from models.Livro import Livro
import os

class LivroService:
    def __init__(self):
        self.livros = self.carregar_livros()

    def carregar_livros(self):
        livros = []
        livros_path = os.path.join("dados", "livros.txt")
        with open(livros_path, "r") as arquivo:
            for linha in arquivo:
                dados = linha.strip().split(";")
                if len(dados) == 3:
                    livros.append(Livro(*dados))
        return livros

    def listar_livros_disponiveis(self, codigo_usuario):
        return [livro for livro in self.livros if not livro.verificar_codigo(codigo_usuario)]

    def cadastrar_livro(self, codigo, nome, autor):
        if any(livro.codigo == codigo for livro in self.livros):
            return False, "Código do livro já existe."
        novo_livro = Livro(codigo, nome, autor)
        self.livros.append(novo_livro)
        return True, "Livro cadastrado com sucesso!"

```

BookWorms
SOFTWARE DEVELOPMENT

Principais Funcionalidades:

Controle de Empréstimos:

- Registro de empréstimos, incluindo prazo de devolução.
- Exibição de empréstimos realizados pelos usuários autenticados.

```python
from models.Emprestimo import Emprestimo
from datetime import date, timedelta
import os

class EmprestimoService:
    def __init__(self):
        self.emprestimos = self.carregar_emprestimos()

    def carregar_emprestimos(self):
        emprestimos = []
        emprestimos_path = os.path.join("dados", "emprestimos.txt")
        with open(emprestimos_path, "r") as arquivo:
            for linha in arquivo:
                dados = linha.strip().split(";")
                if len(dados) == 4:
                    emprestimos.append(Emprestimo(*dados))
        return emprestimos

    def listar_emprestimos(self, codigo_usuario):
        return [e for e in self.emprestimos if e.codigo_usuario == codigo_usuario]

    def realizar_emprestimo(self, codigo_usuario, codigo_livro):
        for emprestimo in self.emprestimos:
            if emprestimo.codigo_usuario == codigo_usuario and emprestimo.codigo_livro == codigo_livro:
                return False, "Você já realizou o empréstimo deste livro."

        novo_emprestimo = Emprestimo(
            str(len(self.emprestimos) + 1),
            codigo_usuario,
            codigo_livro,
            date.today(),
            date.today() + timedelta(days=7)
        )
        self.emprestimos.append(novo_emprestimo)
        return True, "Empréstimo realizado com sucesso!"
```

BookWorms
SOFTWARE DEVELOPMENT

BW Library:

Esse sistema foi projetado para promover organização e acessibilidade no ambiente de bibliotecas, unindo tecnologia e eficiência para fomentar a leitura.

BookWorms
SOFTWARE DEVELOPMENT

Muito Obrigado!