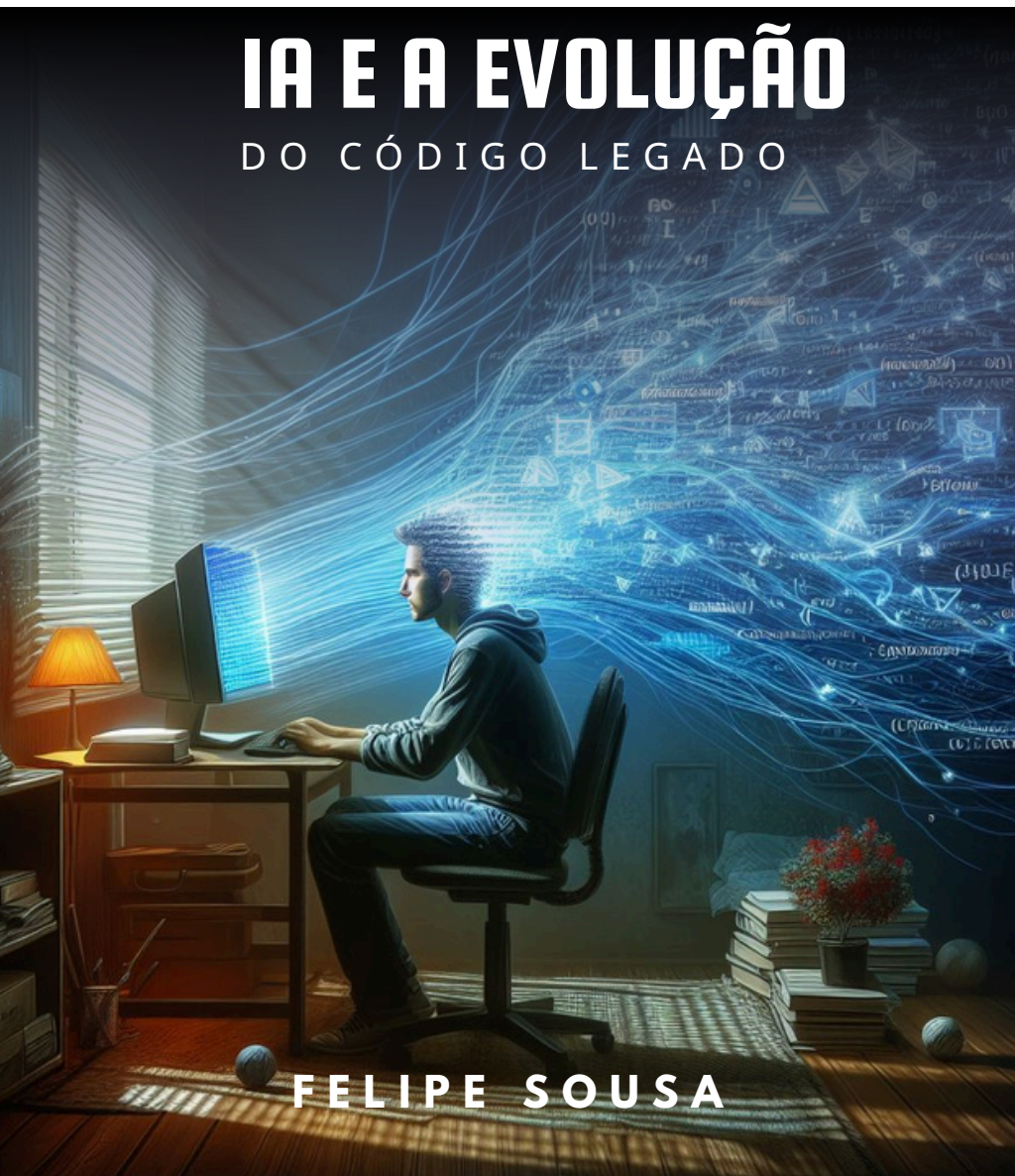


TRANSFORMAÇÃO
INTELIGENTE:

IA E A EVOLUÇÃO DO CÓDIGO LEGADO



FELIPE SOUSA

TRANSFORMAÇÃO

INTELIGENTE:

IA E A EVOLUÇÃO

DO CÓDIGO LEGADO

FELIPE SOUSA

ÍNDICE

Capítulo 1	04
Capítulo 2	05
Capítulo 3	07
Capítulo 4	10
Conclusão	13
Agradecimentos	14



Capítulo I

O Que São Códigos Legados?

Códigos legados são sistemas antigos, muitas vezes essenciais para uma organização, mas que foram escritos em linguagens ou paradigmas que não são mais populares. Manter esses sistemas pode ser desafiador, pois a documentação pode ser escassa e os programadores que os criaram podem não estar mais disponíveis.

Capítulo 2

Desafios dos Códigos Legados

Manter códigos legados apresenta vários desafios:

- **Documentação Deficiente:**
Muitas vezes, os sistemas antigos têm pouca ou nenhuma documentação.
- **Complexidade do Código:**
Códigos legados tendem a ser complexos e difíceis de entender.
- **Falta de Conhecimento:**
Programadores que entendem as linguagens antigas são raros.

Capítulo 2

Exemplo de Código Legado

Vamos considerar um exemplo de código em COBOL, uma linguagem legada comum em sistemas bancários:

```
COBOL

IDENTIFICATION DIVISION.
PROGRAM-ID. exemplo.
DATA DIVISION.
WORKING-STORAGE SECTION.
01 WS-VAR PIC 9(4) VALUE 1000.
PROCEDURE DIVISION.
DISPLAY "VALOR: " WS-VAR.
STOP RUN.
```

Capítulo 3

IA na Manutenção de Códigos Legados

A inteligência artificial (IA) pode ser uma ferramenta poderosa na manutenção de códigos legados. Ela pode ajudar a entender, documentar e até mesmo refatorar o código antigo.

Capítulo 3

Compreensão de Código

A IA pode analisar o código e gerar documentação automaticamente, ajudando a entender como o sistema funciona.

Exemplo com Python

Usando IA, podemos gerar documentação para o código COBOL Apresentado anteriormente:

```
PYTHON

from langchain import CodeToText

cobol_code = """
    IDENTIFICATION DIVISION.
    PROGRAM-ID. exemplo.
    DATA DIVISION.
    WORKING-STORAGE SECTION.
    01 WS-VAR PIC 9(4) VALUE 1000.
    PROCEDURE DIVISION.
    DISPLAY "VALOR: " WS-VAR.
    STOP RUN.
"""

documentation = CodeToText.generate(cobol_code)
print(documentation)
```

Saída Esperada:
Este programa
COBOL exibe o
valor da variável
WS-VAR, que é
inicialmente 1000.

Capítulo 3

Refatoração Automática

A IA pode sugerir melhorias e até mesmo reescrever partes do código em uma linguagem mais moderna.

Exemplo com Java

Vamos converter o código COBOL para Java usando uma ferramenta de IA:

```
Java

public class Exemplo {
    public static void main(String[] args) {
        int valor = 1000;
        System.out.println("VALOR: " + valor);
    }
}
```

Capítulo 4

Uso da IA na otimização de Sistemas Legados

A inteligência artificial (IA) não apenas facilita a manutenção, mas também pode otimizar tarefas e adicionar novas funcionalidades a sistemas legados. Além disso, a IA pode ser usada em áreas como tradução de linguagens e análise de código.


Capítulo 4

Tradução de Linguagens

A IA pode ser usada para traduzir código de uma linguagem legada para uma mais moderna.

Exemplo com PL/SQL para PostgreSQL

Imagine que temos um procedimento PL/SQL que precisa ser convertido para PostgreSQL.



```
PL/SQL

CREATE PROCEDURE exemplo IS
BEGIN
    DBMS_OUTPUT.PUT_LINE('Hello, World!');
END;
```

```
PostgreSQL

CREATE FUNCTION exemplo() RETURNS void AS $$
BEGIN
    RAISE NOTICE 'Hello, World!';
END;
$$ LANGUAGE plpgsql;
```

Capítulo 4

Análise de Código

Ferramentas de IA podem analisar grandes bases de código para identificar problemas de desempenho e segurança.

Exemplo com C

Suponha que temos um código em C que precisa ser analisado:

```
● ● ● C
#include <stdio.h>

void exemplo() {
    int i;
    for(i = 0; i < 10; i++) {
        printf("Valor: %d\n", i);
    }
}
```

Conclusão

A manutenção de códigos legados é um desafio constante, mas a inteligência artificial pode tornar esse processo mais simples e eficiente. Desde a compreensão e documentação até a refatoração e tradução de linguagens, a IA oferece diversas ferramentas que podem ajudar a prolongar a vida útil de sistemas antigos, garantindo que continuem a atender as necessidades das organizações.

Agradecimentos

Quero agradecer por você ter lido este ebook. Este material foi criado com a colaboração de tecnologias de IA para garantir que o conteúdo fosse eficiente e preciso. A diagramação foi feita por um humano, para que o layout ficasse bonito e fácil de ler.

Todas as informações apresentadas foram cuidadosamente revisadas por especialistas humanos, garantindo que tudo fosse preciso e relevante. Espero que você tenha achado este ebook útil e que ele ajude a simplificar e melhorar a manutenção dos seus sistemas legados.

Muito obrigado por dedicar seu tempo a este ebook e por confiar nas minhas orientações.

Crie Seu próprio ebook com IA.

<https://github.com/felipeAguarCode/prompts-recipe-to-create-a-ebook>



Felipe Sousa



github.com/felipeocs/



[linkedin.com/in/felipeocsousa/](https://www.linkedin.com/in/felipeocsousa/)



[instagram.com/eu.felipesousa/](https://www.instagram.com/eu.felipesousa/)