



# **Avaliando a capacidade do ChatGPT desenvolver um sistema de informação real**

**Palavras-Chave: ChatGPT, Desenvolvimento, Software**

**Autores(as):**

**HEITOR PAVANI NOLLA, FT – Unicamp**

**Prof. Dr. Plínio Roberto Souza Vilela(orientador), FT - Unicamp**

---

## **INTRODUÇÃO:**

Desde o lançamento do ChatGPT, no fim de 2022, o programa impressiona com suas capacidades. Usuários informais da ferramenta enaltecem suas habilidades, e especulações surgem, na área da tecnologia, acerca da possibilidade do labor humano ser substituído pela Inteligência Artificial.

Apesar de, informalmente, a ferramenta ser elogiada por seus usuários, cientificamente, nenhum estudo aponta a possibilidade dela ser capaz de substituir programadores humanos.

Nesse contexto, o foco da pesquisa conduzida foi avaliar as capacidades do ChatGPT como desenvolvedor de um sistema de informação. No decorrer do trabalho, resultados interessantes e reveladores acerca das habilidades do ChatGPT na Engenharia de Software foram obtidos, e serão detalhados a seguir.

### **Objetivo da Pesquisa**

O objetivo desta pesquisa foi, inicialmente, analisar a capacidade do ChatGPT de gerar códigos para um sistema de informação completo, com interface gráfica e persistência em banco de dados, e avaliar a viabilidade de integrá-lo a uma equipe de desenvolvimento real.

## **METODOLOGIA:**

Para atingir os objetivos, a pesquisa foi dividida nas seguintes etapas:

### **1 - Desenvolvimento de Sistemas Utilizando ChatGPT:**

Dois sistemas simples de lista telefônica foram desenvolvidos em Java, permitindo adicionar, atualizar, listar e excluir contatos. A interface gráfica foi desenvolvida usando Java Swing.

Um sistema utilizou o banco de dados relacional H2 e o outro, o framework de mapeamento Objeto-Relacional Hibernate.

### **2 - Revisão Bibliográfica Sistemática:**

Uma revisão bibliográfica sistemática foi conduzida, analisando 43 estudos sobre o uso do ChatGPT na engenharia de software.

A revisão buscou embasar o conhecimento obtido durante o desenvolvimento dos sistemas e identificar outras áreas da engenharia de software onde o ChatGPT poderia ser útil.

## RESULTADOS INICIAIS:

O objetivo por trás do desenvolvimento de dois sistemas iguais, porém com diferentes frameworks, estava em analisar a diferença entre o desempenho do ChatGPT ao trabalhar com uma ferramenta familiar ao pesquisador (H2) comparado com uma ferramenta estranha ao pesquisador (Hibernate), desta forma emulando um ambiente de aplicação real, onde o desenvolvedor de software pode ser familiarizado com as ferramentas utilizadas no desenvolvimento ou não. Este esforço se mostrou bastante produtivo pois notou-se que, mesmo para um sistema simples e de fácil desenvolvimento, ainda é necessário que o usuário tenha conhecimentos básicos das ferramentas que está usando para alcançar uma entrega satisfatória.

Além disso, observando as dificuldades geradas ao trabalhar com o ChatGPT em um sistema de pequeno porte, supõe-se que os problemas encontrados seriam ainda maiores conforme os sistemas escalam em tamanho. Durante a revisão bibliográfica conduzida, tal suposição foi corroborada por diversos autores. Assim, condensamos o conhecimento obtido em um artigo: “*Replacing the Human Programmer with ChatGPT: A Systematic Literature Review*” (Substituindo o Programador Humano pelo ChatGPT: Uma Revisão Bibliográfica Sistemática), discutindo as limitações da geração de código com o ChatGPT e identificando outras áreas dentro da Engenharia de Software que poderiam se beneficiar do seu uso.

Este artigo foi finalizado e submetido para publicação em um jornal da *Association for Computing Machinery* (ACM) chamado *Transactions on Software Engineering* (TOSEM), porém foi rejeitado. A partir do feedback recebido dos avaliadores, estamos trabalhando em uma reestruturação do artigo para novo envio em breve.

### Conclusões parciais:

A elaboração do artigo mencionado marcou a finalização do objetivo inicial da pesquisa: analisar a capacidade do ChatGPT de gerar códigos para um sistema de informação completo. Concluiu-se, de maneira geral, que o código gerado pelo ChatGPT apresenta inconsistência notável, vulnerabilidade e qualidade de nível intermediário. Dada sua dependência às instruções fornecidas pelo usuário, os resultados tendem a variar significativamente com base no problema em questão.

Assim, o uso do ChatGPT para geração de código não é defendido. Pelo contrário, a verdadeira proeminência dessa tecnologia emergiu em áreas envolvendo segurança e análise de desempenho do código, bem como em atividades relacionadas à documentação e expressão criativa.

Em geral, a versão atual do ChatGPT (entre meados de 2023 e 2024, versão 3.5) poderia ser bem aplicada em: (i) automação de tarefas consideradas tediosas (como documentação de código, escrita de testes, etc.); (ii) na auto-colaboração no desenvolvimento, agindo como uma espécie de *pair-programmer* para o desenvolvedor humano; (iii) em áreas relacionadas à criatividade, sendo aplicado em sessões de *brainstorming*, e (iv) como um *debugger*.

### Esforços futuros:

Dentro da revisão realizada, uma colocação se destacou:

O autor Adam Hörnemalm, em [1], diz acreditar que a geração de códigos via ChatGPT funcionaria melhor seguindo o modelo *Test-Driven Development* (TDD). Isso, até o momento, não havia sido formalmente testado. Futuramente, ao realizar uma nova iteração da revisão bibliográfica, um novo artigo foi encontrado, [2], propondo o uso do ChatGPT para o desenvolvimento de software seguindo três frameworks, um destes sendo o TDD. Os frameworks aparentavam melhorar

significativamente a qualidade do código final, além de possibilitarem que o ChatGPT trabalhasse em sistemas maiores e mais complexos.

Com isso, dirigimos nossos esforços ao teste dos frameworks propostos no artigo, colocando-os em prática para o desenvolvimento de um novo sistema.

## **Desenvolvimento e Avaliação de um Framework de Desenvolvimento:**

Com base no artigo mencionado, que propõe o uso do ChatGPT nos frameworks Waterfall, Scrum e TDD, um sistema para uma clínica veterinária foi desenvolvido.

Inicialmente, o sistema foi desenvolvido humanamente, a fim de gerar uma *baseline* de comparação. Após isso, o mesmo sistema foi desenvolvido, apenas com códigos gerados pelo ChatGPT. Finalmente, passamos à fase de desenvolvimento do sistema utilizando os frameworks propostos, com foco em testar o desempenho do ChatGPT, a praticidade do uso dos frameworks e a qualidade do software final comparada à versão humana. A hipótese levantada foi: (i) para sistemas com interface gráfica, será necessário propor um *workflow* misto, que integre o desenvolvimento via ChatGPT com o desenvolvedor humano, já que a ferramenta possui grandes dificuldades em gerar, adaptar e refatorar interfaces; (ii) para outros sistemas, os frameworks propostos aplicados à risca, acompanhados de humanos experientes em sua execução, podem gerar resultados satisfatórios em um menor tempo hábil.

### **Dificuldades:**

Essa fase da pesquisa iniciou-se relativamente tarde no cronograma, o que fez com que pouco tempo restasse para seu desenvolvimento. Além disso, dois principais problemas foram encontrados na aplicação deste estudo em um contexto real:

1. A ausência de instruções específicas para cada função do ChatGPT dificultou o desenvolvimento do sistema usando os frameworks propostos. As instruções são bastante vagas e rasas, gerando o segundo problema.
2. O sucesso na aplicação do framework requer conhecimentos avançados em teste de software, engenharia de requisitos, arquitetura e planejamento de sprints.

Essa etapa da pesquisa foi incompleta, e segue como uma forte recomendação de trabalho futuro na área do desenvolvimento integrado ao ChatGPT.

## **RESULTADOS E DISCUSSÃO:**

O desempenho do ChatGPT depende bastante das versões utilizadas. Durante o desenvolvimento desta Iniciação Científica, utilizamos a versão 3.5, a única disponível abertamente. Desde então, a versão 4 foi disponibilizada abertamente de maneira limitada, e aparenta ser uma melhoria significativa em comparação à versão anterior. Além disso, segundo a criadora do modelo, *OpenAI*, a versão 5 está em desenvolvimento, e promete melhorias ainda maiores à ferramenta.

A partir do desenvolvimento do sistema de lista telefônica, a primeira conclusão que obtivemos, posteriormente corroborada por diversos artigos, é que a qualidade do código gerado pelo ChatGPT tende a ser proporcional à qualidade do prompt e à quantidade de informações que lhe é fornecida. O ChatGPT consegue gerar código de qualidade aceitável mas, para isso, o usuário deve conhecer bem o que está pedindo, a ponto de entender a resposta gerada e analisar sua qualidade. Além disso,

neste momento, já ficou claro que o desenvolvimento de interfaces gráficas seria bastante difícil, mas isso seria explorado mais à fundo posteriormente.

A partir da revisão bibliográfica e do artigo produzido, extraímos algumas conclusões: de modo geral, embora o ChatGPT demonstre competência na geração de códigos corretos até determinado nível de complexidade, não existem evidências que apontem a possibilidade da substituição de programadores humanos. Suas áreas de destaque concentram-se, justamente, em áreas nas quais possa servir de apoio a programadores, como: (i) automação de tarefas consideradas tediosas; (ii) colaboração autônoma no desenvolvimento; (iii) domínios relacionados à criatividade e concepção de funcionalidades; e (iv) funções de depuração.

Futuramente, ao desenvolver os sistemas de Clínica Veterinária, tornou-se claro que o desenvolvimento de interfaces gráficas com o ChatGPT é bastante difícil e, de modo geral, não é prático em um ambiente de desenvolvimento. Assim, para sistemas em arquitetura MVC ou similares, um modelo de trabalho mais eficaz seria desenvolver a interface gráfica humanamente, e designar ao ChatGPT tarefas de lógica, arquitetura e geração de códigos para as camadas *Model* e *View*.

No que diz respeito à colaboração autônoma no desenvolvimento, recomendamos como trabalho futuro a continuação do trabalho visto em [1], implementando versões de sistemas, seguindo os diferentes frameworks. Usar o conteúdo presente neste artigo para implementar diferentes versões de um mesmo sistema, compará-los e, para sistemas com interface gráfica ou que necessitem revisão humana, propor um *workflow* misto (IA e Humano) pode ser um bom caminho para acrescentar ao conhecimento científico nesta área.

De modo geral, a perspectiva do uso do ChatGPT na Engenharia de Software é interessante. O consenso acerca da ferramenta parece ser de que o resultado que ela gera tende a variar drasticamente conforme as instruções a ela fornecidas. Assim, torna-se claro que, para extrair o máximo possível do ChatGPT, é necessário conhecer profundamente a ferramenta e possuir boa compreensão acerca do que se está solicitando.

Finalmente, com o surgimento de novos modelos de linguagem de larga escala como Mistral, Llama e Claude, carecem estudos que comparem, entre eles, o desempenho em tarefas similares. Além disso, áreas como Engenharia de Requisitos, Teste de Software e Arquitetura de Software podem ser beneficiadas de estudos mais aprofundados que integrem a elas o uso do ChatGPT, visto que o foco dos estudos aparenta permanecer na área da geração de códigos, na qual, atualmente, um consenso acerca das habilidades do ChatGPT aparenta ter sido atingido.

## BIBLIOGRAFIA

[1] Hörnemalm, A. (2024). *ChatGPT as a software development tool: The future of development* (Supervised by O. Norberg, Examined by T. Mejtoft).

[2] Lin, F., Kim, D. J., & Chen, T.-H. (2024). When LLM-based code generation meets the software development process. Software Performance, Analysis and Reliability (SPEAR) Lab, Concordia University, Montreal, Canada.