



UNIVERSIDADE NOVE DE JULHO - UNINOVE  
PROJETO EM SISTEMAS INTELIGENTES

EXCLUÍDOS OS DADOS SOBRE OS AUTORES EM ATENDIMENTO A  
LGPD - LEI GERAL DE PROTEÇÃO DE DADOS

HealthTech Solutions

São Paulo  
2024

**EXCLUÍDOS OS DADOS SOBRE OS AUTORES EM ATENDIMENTO A  
LGPD - LEI GERAL DE PROTEÇÃO DE DADOS**

**HealthTech Solutions**

Projeto apresentado a Universidade Nove de  
Julho - UNINOVE, como parte dos requisitos obrigatórios para obtenção do título de  
Ciência da computação.

Prof. Orientador: Edson Melo de Souza, Dr.

**São Paulo  
2024**

## RESUMO

---

**Contexto:** O código back-end desenvolvido é uma aplicação de Inteligência Artificial sistêmica que permite o cadastro de novos pacientes e fornece informações como idade e endereço a partir dos dados inseridos pelo usuário. Além disso, o sistema realiza a alocação dos pacientes em leitos hospitalares de acordo com suas necessidades.

Para a obtenção do endereço, o código integra-se a uma API de busca de CEP, que retorna as informações necessárias com base no CEP fornecido. A aplicação foi implementada utilizando o ambiente de desenvolvimento IntelliJ.

Para fins de teste, o sistema possui a capacidade de alocar pacientes em 10 leitos hospitalares disponíveis. Este projeto visa facilitar e agilizar o processo de gerenciamento de pacientes em ambientes hospitalares, contribuindo para uma melhor organização e eficiência dos serviços de saúde.

**Palavras-chave:** Inteligência artificial sistêmica, cadastro de pacientes, alocação de leito.

# ABSTRACT

---

**Contextualization:** The developed back-end code constitutes a systemic AI application enabling the registration of new patients and providing information such as age and address based on user input. Additionally, the system allocates patients to hospital beds according to their needs.

To acquire the address, the code integrates with a ZIP code lookup API, which returns the necessary information based on the provided ZIP code. The application was implemented using the IntelliJ development environment.

For testing purposes, the system can allocate patients to 10 available hospital beds. This project aims to streamline and expedite the patient management process in hospital settings, contributing to improved organization and efficiency in healthcare services..

**Keywords:** Systemic Artificial Intelligence, patient registration, bed allocation

# SUMÁRIO

---

<b>1</b>	<b>Introdução</b>	<b>7</b>
1.1	Contextualização . . . . .	7
1.1.1	Impacto na sociedade . . . . .	7
1.1.2	Costumes e boas práticas . . . . .	7
1.1.3	Origem da ideia . . . . .	7
1.1.4	Tratativas Empresariais com o Cliente . . . . .	7
1.1.5	Sobre os criadores . . . . .	7
1.1.6	Visão e valores . . . . .	8
<b>2</b>	<b>Metodologia</b>	<b>9</b>
2.1	Visão Geral . . . . .	9
2.2	Escolha da linguagem java . . . . .	9
2.3	Ambiente de Desenvolvimento IntelliJ . . . . .	9
2.4	Abordagem de Desenvolvimento dos Scripts . . . . .	10
<b>3</b>	<b>Descrição das atividades</b>	<b>12</b>
3.1	Analista de suporte técnico júnior - Junior e Vitor . . . . .	12
3.2	Desenvolvedor de software pleno - Paulo . . . . .	12
3.3	Desenvolvedor de software sênior - Daniel . . . . .	12
3.4	Product owner - Márcio Galvão . . . . .	13
<b>4</b>	<b>Cronograma</b>	<b>14</b>
4.1	Março: . . . . .	14
4.2	Abril: . . . . .	14
4.3	Maio: . . . . .	15
4.4	Cronograma de Desenvolvimento e Implantação . . . . .	15
4.5	Fase 1: Planejamento e Design . . . . .	15
4.6	Fase 2: Desenvolvimento . . . . .	15
4.7	Fase 3: Testes e Validação . . . . .	15
4.8	Fase 4: Preparação para Lançamento . . . . .	16
4.9	Fase 5: Implantação e Lançamento . . . . .	16
4.10	Fase 6: Pós-Lançamento . . . . .	16
4.11	Fase 7: Avaliação e Manutenção . . . . .	16
4.12	Considerações Finais: . . . . .	16
<b>5</b>	<b>Referências</b>	<b>17</b>
5.1	Referências . . . . .	17

<b>6</b>	<b>Conclusão</b>	<b>18</b>
6.1	Conclusão do projeto . . . . .	18

# 1 INTRODUÇÃO

---

## Resumo do capítulo

*HealthTech Solutions é uma empresa de tecnologia da saúde que desenvolve sistemas de inteligência artificial para otimizar o gerenciamento de leitos hospitalares. Com sede em São Paulo, Brasil, a empresa oferece uma solução back-end que integra cadastro de pacientes, alocação de leitos e verificação de endereços via API de CEP.*

## 1.1 CONTEXTUALIZAÇÃO

### 1.1.1 Impacto na sociedade

O sistema da HealthTech Solutions tem um impacto significativo na sociedade, melhorando a eficiência dos hospitais e a qualidade do atendimento ao paciente. A alocação rápida de leitos pode salvar vidas, e a precisão dos dados de endereço ajuda na logística e no planejamento de recursos.

### 1.1.2 Costumes e boas práticas

A empresa adota uma cultura de inovação e responsabilidade social. As boas práticas incluem desenvolvimento ágil, testes contínuos, transparência nas operações e um forte compromisso com a privacidade e segurança dos dados dos pacientes.

### 1.1.3 Origem da ideia

A ideia surgiu em 2023, quando os fundadores perceberam a necessidade de um sistema que pudesse lidar com a complexidade do gerenciamento de leitos hospitalares de forma mais eficiente, utilizando o poder da IA para automatizar e otimizar o processo.

### 1.1.4 Tratativas Empresariais com o Cliente

A HealthTech Solutions aborda as tratativas empresariais com foco no cliente, oferecendo suporte personalizado, treinamento e acompanhamento contínuo para garantir a satisfação e o melhor uso do sistema.

### 1.1.5 Sobre os criadores

Os criadores da empresa são João Silva, Maria Costa e Luiz Pereira, profissionais com experiência em tecnologia e saúde. Eles compartilham uma visão de transformar o setor de saúde por meio da tecnologia e trabalham juntos para tornar essa visão uma

realidade

#### 1.1.6 Visão e valores

**Visão:** Ser líder global em soluções de tecnologia da saúde, transformando o gerenciamento de leitos hospitalares através da inovação e inteligência artificial. Nosso objetivo é melhorar a eficiência dos hospitais e a qualidade do atendimento ao paciente, garantindo que cada indivíduo receba os cuidados necessários de forma rápida e humanizada

**Valores:** Buscamos constantemente avanços tecnológicos para superar os desafios do setor de saúde.

1. **Excelência:** Comprometemo-nos com a mais alta qualidade em nossos produtos e serviços.
2. **Integridade:** Agimos com honestidade e transparência em todas as nossas operações.
3. **Colaboração:** Valorizamos o trabalho em equipe e a parceria com clientes, funcionários e stakeholders.
4. **Empatia:** Colocamos as necessidades dos pacientes no centro de tudo o que fazemos.
5. **Responsabilidade Social:** Contribuímos para uma sociedade mais saudável e acessível a todos

Estes são os pilares que orientam a HealthTech Solutions em sua missão de revolucionar o cuidado ao paciente e otimizar os processos hospitalares.



## 2 METODOLOGIA

---

### Resumo do capítulo

*HealthTech Solutions: Argumentação sobre Linguagem e Desenvolvimento de Scripts.*

#### 2.1 VISÃO GERAL

Na HealthTech Solutions, cada decisão de tecnologia é cuidadosamente considerada para garantir que nossos sistemas atendam aos mais altos padrões de desempenho, segurança e escalabilidade. Ao desenvolver o sistema de gestão hospitalar, escolhemos Java como a linguagem principal para o back-end, e o ambiente de desenvolvimento IntelliJ como nossa ferramenta de escolha. Aqui estão os motivos para essas escolhas e a abordagem adotada para o desenvolvimento dos scripts:

#### 2.2 ESCOLHA DA LINGUAGEM JAVA

Java é uma linguagem conhecida por sua robustez e confiabilidade. Sua arquitetura orientada a objetos e seu sistema de gerenciamento de memória automático fornecem um ambiente seguro e estável para o desenvolvimento de aplicativos de missão crítica, como sistemas de gestão hospitalar.

1. **Escalabilidade:** Java oferece uma excelente escalabilidade, permitindo que nossos sistemas cresçam conforme as demandas dos clientes e do mercado. Com a ampla adoção de Java em empresas de todos os tamanhos, podemos ter confiança de que nossa solução será capaz de lidar com grandes volumes de dados e tráfego de usuários
2. **Ecosistema e Suporte:** Java possui um vasto ecossistema de ferramentas, frameworks e bibliotecas que facilitam o desenvolvimento de aplicativos complexos. Frameworks como Spring Framework fornecem funcionalidades robustas para o desenvolvimento de aplicativos empresariais, enquanto bibliotecas como Apache HttpClient simplificam a integração com serviços externos.
3. **Portabilidade:** Java é conhecido por sua portabilidade, o que significa que nossos sistemas podem ser executados em uma ampla variedade de plataformas e ambientes. Isso nos dá a flexibilidade de implantar nossas soluções em diferentes infraestruturas, desde servidores locais até a nuvem

#### 2.3 AMBIENTE DE DESENVOLVIMENTO INTELIJ

IntelliJ é reconhecido por sua alta produtividade e conjunto abrangente de recursos que aceleram o desenvolvimento de software. Sua interface intuitiva, refatorações inte-

ligentes e sugestões de código tornam o processo de desenvolvimento mais eficiente e agradável.

1. **Integração com Ferramentas:** IntelliJ oferece uma integração perfeita com outras ferramentas e tecnologias, facilitando o desenvolvimento em um ambiente integrado. Desde a integração com sistemas de controle de versão até a depuração e teste de aplicativos.
2. **Suporte à Linguagem Java:** Como uma IDE voltada principalmente para desenvolvedores Java, IntelliJ oferece um suporte excepcional à linguagem, incluindo destaque de sintaxe, autocompletar inteligente, análise estática de código e muito mais. Isso permite que nossos desenvolvedores escrevam código Java limpo e eficiente com facilidade.
3. **Compatibilidade com Diversos Frameworks e Bibliotecas:** O IntelliJ suporta uma ampla gama de frameworks e bibliotecas populares em Java, como Spring, Hibernate e JUnit. Isso permite que os desenvolvedores trabalhem eficientemente com essas ferramentas dentro do ambiente do IntelliJ.
4. **Comunidade Forte e Ativa:** Tanto o Java quanto o IntelliJ têm uma comunidade de desenvolvedores forte e ativa. Isso significa que, se os desenvolvedores encontrarem problemas ou tiverem perguntas, é provável que possam encontrar respostas e soluções na comunidade.
5. **Facilidade de Uso:** O IntelliJ é conhecido por sua interface de usuário amigável e fácil de navegar. Ele oferece uma variedade de atalhos de teclado que podem aumentar significativamente a velocidade de codificação. Além disso, o IntelliJ fornece assistência de código em tempo real, o que pode ajudar a prevenir e corrigir erros rapidamente.

## 2.4 ABORDAGEM DE DESENVOLVIMENTO DOS SCRIPTS

Seguimos uma abordagem de arquitetura modular para garantir a coesão e a reutilização do código. Dividimos o sistema em componentes independentes, cada um responsável por uma funcionalidade específica, facilitando a manutenção e evolução do sistema ao longo do tempo.

1. **Padrões de Design:** Adotamos padrões de design comprovados, como MVC (Model-View-Controller), para separar as preocupações e promover a escalabilidade e a flexibilidade do sistema. Isso nos permite construir um sistema bem estruturado e fácil de entender, facilitando a colaboração entre os membros da equipe de desenvolvimento

2. **Testes Automatizados:** Implementamos testes automatizados para garantir a qualidade e a estabilidade do código. Utilizamos frameworks de teste como JUnit e Mockito para escrever testes unitários e de integração que validam o comportamento do sistema em diferentes cenários.
3. **Documentação Abrangente:** Mantemos uma documentação abrangente do código e da arquitetura do sistema, facilitando o entendimento e a manutenção por parte dos desenvolvedores. Isso inclui documentação de código, diagramas de arquitetura, guias de contribuição e muito mais.

Em resumo, a escolha de Java como linguagem de programação e o uso do ambiente de desenvolvimento IntelliJ são fundamentais para o sucesso de nossos projetos na HealthTech Solutions. Com uma abordagem cuidadosa e uma equipe talentosa de desenvolvedores, estamos confiantes de que podemos fornecer soluções de alta qualidade que atendam às necessidades de nossos clientes e contribuam para o avanço da saúde digital.

## 3 DESCRIÇÃO DAS ATIVIDADES

---

### 3.1 Analista de suporte técnico junior - JUNIOR E VITOR

#### Atividades:

1. Presta assistência técnica aos usuários, resolvendo problemas de software e hardware.
2. Mantém a infraestrutura de TI da empresa, garantindo a operação contínua dos sistemas.
3. Documenta procedimentos técnicos e cria manuais de usuário.
4. Colabora com a equipe de desenvolvimento para identificar e corrigir bugs.

### 3.2 Desenvolvedor de software pleno - PAULO

#### Atividades:

1. Desenvolve e mantém aplicações de software conforme as necessidades da empresa.
2. Escreve código limpo e eficiente em várias linguagens de programação.
3. Realiza testes unitários e de integração para garantir a qualidade do software.
4. Participa de reuniões de planejamento e revisão de sprints.

### 3.3 Desenvolvedor de software sênior - DANIEL

#### Atividades:

1. Lidera projetos de desenvolvimento de software, desde a concepção até a entrega.
2. Arquiteta soluções de software escaláveis e de alta disponibilidade.
3. Orienta desenvolvedores júnior e pleno, promovendo o crescimento técnico da equipe.
4. Avalia novas tecnologias e ferramentas para adoção pela empresa.

### 3.4 Product owner - MÁRCIO GALVÃO

#### **Atividades:**

1. Define a visão do produto e gerencia o backlog do produto.
2. Prioriza as funcionalidades e colabora com a equipe de desenvolvimento para entregar valor.
3. Serve como ponto de contato principal para stakeholders e clientes.
4. Analisa o feedback do mercado para aprimorar o produto continuamente.

## 4 CRONOGRAMA

---

### 4.1 Março:

#### **Semana 1: Definição e Planejamento.**

Iniciamos com uma sessão de brainstorming, reunindo nossa equipe para explorar diversas ideias e consolidá-las em uma visão unificada. Utilizamos as referências fornecidas pelo professor como base para nossa discussão.

#### **Semana 2: Transição de ideias**

Iniciamos a transição das ideias do estágio teórico para o prático, optando pela linguagem Java devido à sua relevância e complexidade no mercado de trabalho. Esta escolha foi embasada em sua robustez e flexibilidade para o escopo do projeto.

#### **Semana 3: Adoção do IntelliJ**

Decidimos adotar o IntelliJ como ambiente de desenvolvimento integrado (IDE), reconhecendo sua capacidade de acelerar o processo de desenvolvimento e aumentar nossa eficiência. A integração com o IntelliJ também nos proporcionaria vantagens adicionais, como facilidade na depuração e organização do código.

### 4.2 Abril:

#### **Semana 4: Implementações**

Nesta fase, concentramo-nos na implementação das classes principais do projeto. Realizamos revisões contínuas do código e ajustes conforme necessário, acompanhados de testes unitários básicos para garantir a funcionalidade. Além disso, iniciamos a fase de testes de recursos para validar a integridade do sistema.

#### **Semana 5: Testes Iniciais**

Dedicamos esta semana para testar exaustivamente as funcionalidades do código, identificando e corrigindo quaisquer erros ou inconsistências. Elaboramos um cronograma detalhado para monitorar nosso progresso e garantir que permanecêssemos dentro do prazo estabelecido.

#### **Semana 6: Testes finais**

Finalizamos a implementação do código e conduzimos testes finais para verificar a estabilidade e funcionalidade do aplicativo. Após confirmar que todas as funcionalidades estavam operacionais e atendiam aos requisitos do projeto, concluímos que o aplicativo está pronto para o lançamento oficial aos usuários.

### 4.3 Maio:

#### **Semana 7: Conclusão de Testes**

Realização de testes unitários nas classes Paciente e Leito para assegurar a funcionalidade dos métodos. Teste do fluxo de cadastro de pacientes e alocação em leitos no sistema. Identificação e correção de eventuais bugs e inconsistências nas classes Paciente e Leito.

#### **Semana 8: Ajustes e Melhorias**

Revisão do código das classes Paciente e Leito e otimização de algoritmos, se necessário. Implementação de tratamento de exceções nas classes Paciente e Leito. Refinamento da interface de usuário e experiência do usuário relacionados aos dados do paciente e leitos.

### 4.4 Cronograma de Desenvolvimento e Implantação

#### 4.5 Fase 1: Planejamento e Design

**Mês 1 (Janeiro, semana 1):** Concepção e planejamento do projeto. Reuniões iniciais para definir escopo e objetivos. Pesquisa de mercado e análise de requisitos.

**Mês 2 (Fevereiro, semana 2,3 e 4):** Design do sistema e arquitetura. Criação de wireframes e protótipos de interface do usuário. Definição da arquitetura do sistema e seleção de tecnologias.

#### 4.6 Fase 2: Desenvolvimento

**Mês 3-4 (Março-Abril, semana 5 e 6):** Desenvolvimento do back-end. Configuração do servidor, banco de dados e integração de API de CEP. Implementação do cadastro de pacientes e alocação de leitos.

**Mês 5 (Maio, semana 7,8 e 9):** Desenvolvimento do front-end. Criação de interfaces de usuário e painéis de controle. Integração front-end com serviços back-end.

#### 4.7 Fase 3: Testes e Validação

**Mês 6 (Junho, semana 10 e 11):** Testes unitários e de integração. Execução de testes automatizados e correção de bugs. Revisão de código e otimização de desempenho.

**Mês 7 (Julho, semana 12,13,14 e 15):** Testes de usuário e feedback. Testes beta com um grupo selecionado de usuários. Coleta e implementação de feedback para melhorias.

#### **4.8 Fase 4: Preparação para Lançamento**

**Mês 8 (Agosto, semana 16 e 17):** Preparação para lançamento. Finalização da documentação técnica e manuais do usuário. Estratégias de marketing e comunicação com stakeholders.

#### **4.9 Fase 5: Implantação e Lançamento**

**Mês 9 (Setembro, semana 18,19 e 20):** Implantação e lançamento no mercado. Configuração do ambiente de produção e implantação do sistema. Lançamento oficial e campanha de marketing.

#### **4.10 Fase 6: Pós-Lançamento**

**Mês 10 (Outubro, semana 21, 22 e 23):** Monitoramento e suporte pós-lançamento. Monitoramento contínuo do sistema e suporte técnico. Coleta de dados de uso e feedback dos usuários finais.

#### **4.11 Fase 7: Avaliação e Manutenção**

**Mês 11-12 (Novembro-Dezembro, semana 24 e 25):** Avaliação e manutenção contínua. Análise de desempenho do sistema e identificação de áreas de melhoria. Atualizações regulares do sistema e lançamento de novas funcionalidades.

#### **4.12 Considerações Finais:**

O cronograma é adaptável e pode ser ajustado com base na evolução do projeto.

A comunicação constante com os clientes e a equipe interna é essencial para o sucesso do projeto.

A HealthTech Solutions se compromete a manter altos padrões de qualidade e segurança em todas as fases do projeto.



## 5 REFERÊNCIAS

---

### 5.1 Referências

1. Desenvolvedor JetBrains Lançamento 1.0 / janeiro de 2001 (23 anos) Versão estável 2024.1.1 Escrito em Java, Kotlin Sistema operacional Windows, macOS, Linux Gênero(s) Java IDE Licença Apache 2
2. São Paulo; 2018. PMI - PROJECT MANAGEMENT INSTITUTE. Guia PMBOK®: Um Guia para o Conjunto de Conhecimentos em Gerenciamento de Projetos, Sexta edição, Pennsylvania: PMI, 2017.
3. Desenvolvedor Sun Microsystems (adquirida pela Oracle Corporation) Lançamento Inicial 23 de maio de 1995 (29 anos) Versão Estável Java SE 20 (lançada em março de 2023) Escrito em C, C++ Sistema Operacional Multiplataforma (Windows, macOS, Linux, Solaris, e outros) Gênero(s) Linguagem de Programação Licença GNU General Public License (GPL) com Classpath Exception

## 6 CONCLUSÃO

---

### 6.1 Conclusão do projeto

O projeto de desenvolvimento de um sistema de Inteligência Artificial para gerenciamento de pacientes em ambientes hospitalares, integrando funcionalidades de cadastro, obtenção de informações e alocação em leitos, representa uma solução inovadora e eficaz para otimizar os serviços de saúde. A utilização de tecnologias modernas, como integração com API de busca de CEP e desenvolvimento no ambiente IntelliJ, demonstra o compromisso com a excelência técnica. Além disso, a capacidade de alocação em leitos hospitalares contribui significativamente para a organização e eficiência dos serviços, destacando a relevância e o impacto positivo desse projeto na área da saúde.