UNIVERSIDADE DO CONTESTADO – UnC

CURSO DE ENGENHARIA DE SOFTWARE

CAUÃ FILIPE GRAFF

**TITULO**

CONCÓRDIA

2025

CAUÃ FILIPE GRAFF

**TITULO**

Trabalho de Conclusão de Curso apresentado como exigência para obtenção de nota na graduação de Engenharia de Software, ministrado na Universidade do Contestado – UnC, Campus Concórdia, sob Orientação do (a) Professor (a) Moacir Solano Kichel.

**LISTA DE FIGURAS**

**Nenhuma entrada de sumário foi encontrada.**

**LISTA DE ABREVIATURAS E/OU SIGLAS**

**SUMÁRIO**

[1. INTRODUÇÃO 5](#_Toc193218172)

[1.2. OBJETIVOS 6](#_Toc193218173)

[1.3. Objetivo Geral 6](#_Toc193218174)

[1.4. Objetivos Específicos 6](#_Toc193218175)

[2 REFERENCIAL TEÓRICO 7](#_Toc193218176)

[2.1 PSICOLOGIA ORGANIZACIONAL E DO TRABALHO 7](#_Toc193218177)

[2.1.1 Histórico 7](#_Toc193218178)

[2.2 ESTRESSE NO TRABALHO 8](#_Toc193218179)

[2.2.1 Doenças Provocadas pelo Estresse no Trabalho 9](#_Toc193218180)

[2.3 ESTRESSE NA POLÍCIA MILITAR 9](#_Toc193218181)

[3 MATERIAL E MÉTODOS 12](#_Toc193218182)

[3.1 TIPO DE PESQUISA 12](#_Toc193218183)

[3.2 SUJEITOS DA PESQUISA 12](#_Toc193218184)

[3.3 ASPECTOS ÉTICOS 12](#_Toc193218185)

[3.4 COLETA DE DADOS 13](#_Toc193218186)

[4 RESULTADOS E DISCUSSÕES 14](#_Toc193218187)

[5 CONCLUSÃO 16](#_Toc193218188)

[REFERÊNCIAS 17](#_Toc193218189)

1. INTRODUÇÃO

A transformação digital tem impulsionado diversas áreas do conhecimento, incluindo a contabilidade, que cada vez mais adota ferramentas tecnológicas para otimizar processos e minimizar erros operacionais. O grande volume de informações geradas por empresas, especialmente no que se refere à gestão de ponto eletrônico e controle de jornada de funcionários, demanda soluções eficientes para organização e integração desses dados em sistemas contábeis. Conforme Padoveze (2019), a automação de processos contábeis não apenas reduz falhas humanas, mas também contribui para a eficiência e segurança das informações tratadas. De acordo com Rezende (2018), a digitalização e automação dessas tarefas são fundamentais para aumentar a produtividade e melhorar a qualidade das informações processadas

Atualmente, muitos departamentos contábeis ainda realizam manualmente a importação de dados de frequência de funcionários, o que pode levar a inconsistências fiscais e trabalhistas. Além disso, os formatos variados de arquivos utilizados pelas empresas, **como *Portable Document File* ou Arquivo de Documento Portáil (PDF), *Comma-Separeted-Values* ou Valores Separados por Virgula (CSV)** e planilhas eletrônicas, dificultam a padronização e integração desses dados com sistemas de *Enterprise Resource Planning* ou Planejamento de Recursos Empresarias (ERP).

Segundo Laudon e Laudon (2021), os sistemas ERP são plataformas integradas que automatizam processos de negócios em diversas áreas de uma organização, como finanças, recursos humanos e contabilidade, promovendo uma base de dados única e a padronização das informações. Esses sistemas permitem que empresas tenham maior controle sobre suas operações e otimizem a tomada de decisão com base em dados consistentes e em tempo real.

Para o desenvolvimento do protótipo proposto, serão utilizadas tecnologias como Python, devido à sua robustez em processamento de dados e extração de informações de documentos estruturados e não estruturados (McKinney, 2018), além de bibliotecas como Pandas para manipulação de dados, PyPDF2 para leitura de arquivos PDF e OpenPyXL para interação com planilhas do Excel. Essas ferramentas são amplamente reconhecidas pela comunidade de ciência de dados por sua flexibilidade e eficiência no tratamento de grandes volumes de dados (VanderPlas, 2016).  
 Para o frontend será desenvolvido com o framework React.js, permitindo uma interface interativa e responsiva. Já o backend será construído com o framework FastAPI, que oferece rapidez na criação de APIs RESTful em Python. O banco de dados utilizado será o PostgreSQL, escolhido por sua robustez, confiabilidade e recursos avançados, sendo ideal para aplicações que exigem maior desempenho e escalabilidade.

Nesse contexto, este trabalho propõe o desenvolvimento de um protótipo capaz de extrair, organizar e estruturar dados provenientes de arquivos de ponto, convertendo-os para um layout padronizado e adequado à importação em sistemas ERP. A implementação desse protótipo busca reduzir a necessidade de intervenções manuais, garantindo maior confiabilidade e eficiência no tratamento das informações. Além disso, Martins (2020) destaca que a adoção de tecnologias inovadoras permite que profissionais contábeis se concentrem em atividades estratégicas, deixando tarefas repetitivas e operacionais para sistemas automatizados.

Outro fator relevante é a economia de tempo e recursos. De acordo com Santos e Almeida (2022), empresas que adotam soluções tecnológicas para integração de dados conseguem reduzir o tempo gasto em tarefas operacionais em até 40%, permitindo que os profissionais direcionem esforços para análises mais estratégicas e consultivas. Além disso, a automação contribui para a redução de erros humanos, garantindo maior confiabilidade na geração de relatórios contábeis e fiscais.

A contabilidade moderna exige ferramentas que permitam a integração de diferentes fontes de dados de maneira rápida e precisa, eliminando a necessidade de processos manuais demorados e suscetíveis a falhas. Conforme aponta Silva (2021), empresas que não investem em automação tendem a enfrentar dificuldades na gestão de informações, o que pode impactar diretamente na tomada de decisão e na eficiência dos processos internos.

A gestão do ponto eletrônico é uma obrigação legal regida por normas trabalhistas, como a Portaria 671/2021 do Ministério do Trabalho, que estabelece critérios para o registro da jornada de trabalho. De acordo com Souza e Mendes (2020), o controle de ponto é essencial para assegurar o cumprimento de direitos trabalhistas e evitar passivos judiciais. No entanto, a diversidade de sistemas e formatos de exportação desses dados pode dificultar sua integração com sistemas contábeis, tornando a automação uma necessidade estratégica.

Dessa forma, a implementação de um sistema automatizado para organização e importação de dados de ponto pode trazer impactos positivos não apenas para empresas contábeis, mas também para organizações que precisam gerenciar essas informações de maneira eficiente. Segundo Costa e Ribeiro (2023), a adoção de ferramentas digitais para gestão de dados permite maior conformidade com legislações trabalhistas, assegurando que informações como horas trabalhadas, adicional noturno e horas extras sejam calculadas corretamente e sem falhas.

A justificativa para este estudo reside na necessidade crescente das empresas de automatizar processos internos, garantindo maior agilidade e precisão na gestão contábil. Segundo Oliveira e Silva (2021), a digitalização dos processos contábeis melhora a conformidade fiscal e reduz riscos associados ao manuseio manual de informações sensíveis. A implementação desse protótipo pode trazer impactos positivos para empresas e escritórios de contabilidade, garantindo maior eficiência e precisão na gestão de dados.

A utilização de ferramentas como Python, Pandas, PyPDF2, OpenPyXL, FastAPI, PostgreSQL e React.js é suficiente para consolidar a proposta de um protótipo capaz de automatizar a extração, tratamento e padronização dos dados.

## OBJETIVOS

### Objetivo Geral

Desenvolver um protótipo funcional capaz de automatizar a extração, organização, tratamento e padronização dos dados de ponto eletrônico para posterior importação no sistema ERP, utilizando tecnologias como Python, PostgreSQL, FastAPI e React.js.

### Objetivos Específicos

- Realizar uma revisão bibliográfica sobre os seguintes temas: sistemas de folha de ponto eletrônico, Engenharia de Software, prototipação, modelagem UML, banco de dados e ferramentas de desenvolvimento;

- Efetivar o levantamento dos requisitos do sistema

- Representar os modelos de dados e do sistema utilizando UML.

- Desenvolver um protótipo funcional capaz de automatizar a extração e tratamento dos dados, Python, PostgreSQL.

Realizar testes de funcionamento e utilização do protótipo.

# 2 REFERENCIAL TEÓRICO

**2.1. Sistemas de Controle de Ponto Eletrônico**

Os sistemas de controle de ponto eletrônico são ferramentas tecnológicas utilizadas para registrar e monitorar a jornada de trabalho dos colaboradores em uma organização. Seu principal objetivo é garantir o cumprimento da carga horária estabelecida, bem como registrar entradas, saídas, intervalos e horas extras de forma segura, precisa e automatizada. Esses registros são fundamentais tanto para o cumprimento da legislação trabalhista quanto para a geração de dados que subsidiem o setor contábil e de recursos humanos.

Segundo Souza e Mendes (2020), o ponto eletrônico é um recurso indispensável na administração moderna, pois contribui para a redução de fraudes, aumenta a confiabilidade das informações e facilita a auditoria dos dados registrados. Além disso, ele representa uma alternativa mais eficiente e segura em comparação aos métodos manuais, como livros de ponto e cartões mecânicos, que estão suscetíveis a erros e manipulações.

A legislação brasileira regula o uso do ponto eletrônico por meio da Portaria nº 671/2021, do Ministério do Trabalho e Previdência, que estabelece regras para o registro da jornada de trabalho por meio de sistemas alternativos de controle. De acordo com essa norma, os empregadores podem utilizar diferentes tipos de registro eletrônico, como o Registrador Eletrônico de Ponto Convencional (REP-C), o Alternativo (REP-A) e o por Programa (**REP-P)**, desde que cumpram os requisitos legais e garantam a integridade e a inviolabilidade das informações.

A diversidade de sistemas disponíveis no mercado, bem como os variados formatos de exportação de dados, representa um desafio significativo para a integração dessas informações com sistemas contábeis. Os registros de ponto são frequentemente exportados em formatos como PDF, CSV e planilhas, os quais nem sempre seguem uma estrutura padronizada. Essa falta de uniformidade dificulta a importação direta para sistemas de folha de pagamento e ERP, exigindo, muitas vezes, intervenção manual para ajustes e correções.

Costa e Ribeiro (2023) destacam que a ausência de integração entre os sistemas de controle de ponto e os sistemas contábeis pode gerar atrasos no processamento da folha de pagamento, erros nos cálculos de horas extras e adicionais, além de impactar negativamente na conformidade com obrigações trabalhistas. Esses fatores tornam a automação e a padronização do fluxo de dados entre os sistemas uma necessidade estratégica para organizações que buscam maior eficiência operacional e segurança jurídica.

Além disso, é importante destacar a existência de sistemas legados de ponto eletrônico, ou seja, softwares antigos ainda em uso por diversas organizações, muitas vezes desenvolvidos sob tecnologias obsoletas e com estruturas rígidas de dados. Conforme Oliveira e Cruz (2020), esses sistemas, embora ainda operacionais, apresentam limitações significativas quanto à integração com plataformas modernas, dificultando a extração e o tratamento automatizado das informações. A falta de compatibilidade com padrões atuais de exportação de dados, bem como a ausência de APIs ou interfaces programáveis, impõe desafios adicionais para a automação e a padronização dos registros de jornada.

Dessa forma, observa-se que os sistemas de ponto eletrônico desempenham um papel essencial na gestão de pessoas e no cumprimento das normas legais. Entretanto, sua real efetividade depende da capacidade das empresas em integrar esses dados com outras plataformas corporativas, evitando retrabalho, aumentando a confiabilidade das informações e reduzindo o risco de inconformidades fiscais e trabalhistas.

**2.2. Engenharia de Software**

A Engenharia de Software é uma área da computação voltada ao desenvolvimento sistemático, disciplinado e quantificável de softwares. Segundo Pressman (2016), trata-se da aplicação de uma abordagem estruturada ao desenvolvimento, operação e manutenção de software, com foco na qualidade, produtividade e na redução de custos e riscos. Essa disciplina abrange processos, métodos, ferramentas e práticas que auxiliam na construção de soluções tecnológicas confiáveis e eficientes.

De acordo com Sommerville (2011), a Engenharia de Software envolve atividades como especificação de requisitos, projeto, codificação, testes e manutenção de sistemas. A adoção dessas práticas permite maior controle e previsibilidade no desenvolvimento, favorecendo a criação de sistemas que atendam às necessidades reais dos usuários.

Os processos de software podem seguir diferentes modelos de ciclo de vida, como a cascata, incremental, espiral ou ágil, cada um adequado a contextos e projetos específicos. A escolha do modelo depende da complexidade do sistema, dos requisitos dos usuários e do tempo disponível para o desenvolvimento.

**2.3. Prototipação**

A prototipação é uma abordagem amplamente utilizada na Engenharia de Software para desenvolver versões preliminares de um sistema, com o objetivo de validar funcionalidades, levantar requisitos e coletar feedback de usuários antes da construção final do produto. Essa técnica permite visualizar, testar e ajustar funcionalidades essenciais em ciclos iterativos, promovendo uma maior aderência às necessidades reais dos usuários.

De acordo com Sommerville (2011), a prototipação é essencial para o desenvolvimento incremental de sistemas, pois facilita a comunicação entre usuários e desenvolvedores, especialmente em contextos onde os requisitos são incertos ou mutáveis. Protótipos podem ser de baixa fidelidade como esboços ou de alta fidelidade com partes funcionais já implementadas.

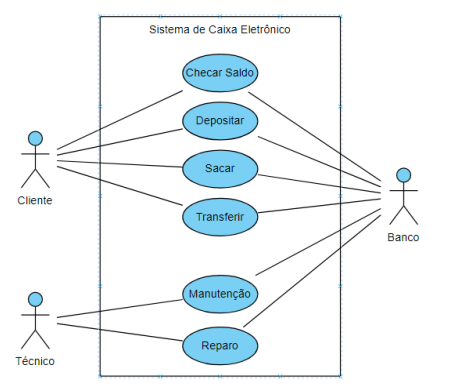
No presente trabalho, a prototipação será aplicada na criação de uma interface funcional que represente o processo de extração, organização e padronização de dados de ponto eletrônico, com o objetivo de validar os requisitos antes da implementação definitiva do sistema.

**2.4. Modelagem UML**

A UML (Unified Modeling Language) é uma linguagem de modelagem padronizada que permite representar graficamente os aspectos estruturais e comportamentais de um sistema de software. Seu principal objetivo é fornecer uma forma visual de documentar, projetar e comunicar a arquitetura de sistemas de forma clara e compreensível para desenvolvedores, analista. Segundo Larman (2007), a UML é uma ferramenta essencial para o processo de desenvolvimento de software, pois facilita a validação dos requisitos e o planejamento da implementação.

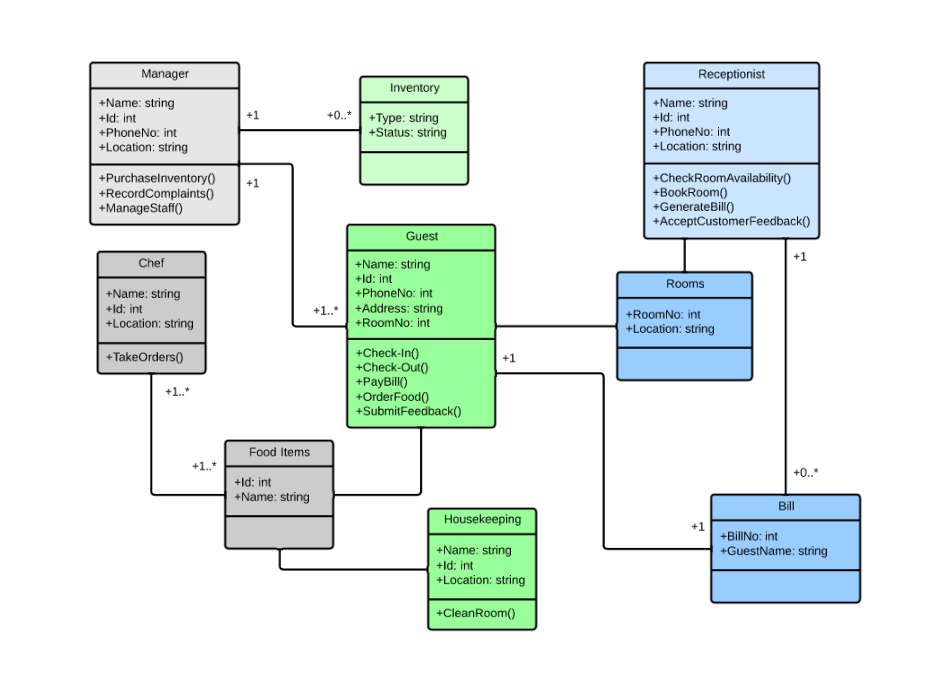
No contexto deste trabalho, a UML será utilizada para representar os principais componentes e funcionalidades do sistema proposto, garantindo maior clareza e organização no projeto. Tendo seu principais diagramas Caso de uso, Classes e Atividades.

Figura 1 – Diagrama de Caso de uso

Fonte: linkedin (2024)

Como apresentado na figura 1, um diagrama de caso de uso é um tipo de diagrama na UML que descreve as funcionalidades ou recursos que um sistema oferece do ponto de vista dos usuários. Ele mostra as interações entre os atores e o sistema em si, destacando os diferentes casos de uso ou cenários de uso que descrevem as operações que o sistema pode realizar em resposta a entradas dos atores. Esses diagramas são frequentemente usados na fase de análise para capturar os requisitos do sistema de uma forma compreensível para todas as partes interessadas.

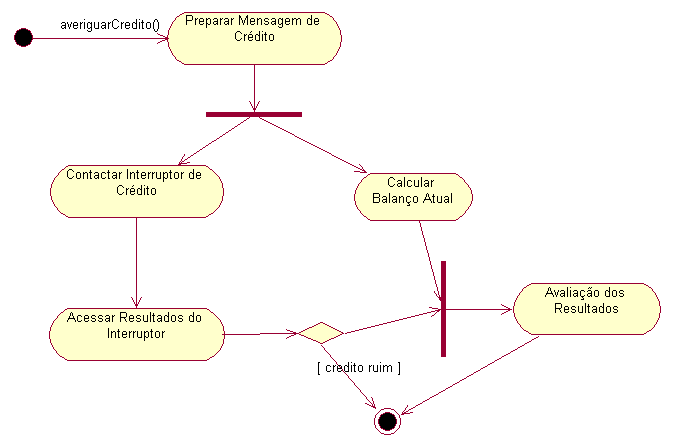
Figura 2 – Diagrama de Classes



Fonte: lucidchart (2025)

O Diagrama de Classes representa a estrutura estática do sistema conforme representado na figura 2, exibindo as classes, seus atributos, métodos e os relacionamentos entre elas. Esse tipo de diagrama é essencial para projetar a arquitetura lógica do sistema e orientar a programação orientada a objetos. De acordo com Pressman (2016), o diagrama de classes fornece uma visão detalhada das entidades do sistema e seus vínculos, como herança, associação e composição.

Figura 2 – Diagrama de Classes



Fonte: ufmg (2025)

O Diagrama de Atividades representa o fluxo de trabalho ou a lógica de execução de um processo dentro do sistema, modelando decisões, ações e paralelismos. Ele é particularmente útil para descrever o comportamento dinâmico de funcionalidades, como a execução de um processo do início ao fim, passo a passo.

**2.5. Banco de Dados**

Os bancos de dados são estruturas essenciais para o armazenamento, organização e recuperação eficiente de informações. Eles são projetados para manter dados consistentes, acessíveis e seguros em sistemas computacionais. Segundo Elmasri e Navathe (2011), um banco de dados é uma coleção de dados inter-relacionados, organizada de forma lógica e estruturada, com o objetivo de atender às necessidades de uma aplicação específica.

Para que esses dados possam ser manipulados de maneira eficaz, utiliza-se um Sistema de Gerenciamento de Banco de Dados (SGBD), que é um conjunto de programas responsáveis por gerenciar o acesso, inserção, exclusão, atualização e recuperação dos dados armazenados. De acordo com Date (2004), os SGBDs também são responsáveis por garantir propriedades fundamentais de segurança, integridade e concorrência, especialmente em ambientes multiusuários.

Existem diversos tipos de banco de dados, dentre eles: banco de dados relacionais e não relacionais, distribuídos, de memória, SQL, NoSQL. Cada um deles tem vantagens e desvantagens, sendo cada um útil com base na situação em que se encontra.

**2.6. Linguagem e Ferramentas de Desenvolvimento**

As linguagens de programação e ferramentas tecnológicas desempenham papel fundamental no desenvolvimento de sistemas informatizados, pois permitem desde a construção da lógica de negócio até a apresentação visual ao usuário. As linguagens de programação são instruções formais utilizadas por desenvolvedores para criar software, aplicativos e sistemas. Entre as mais utilizadas atualmente, destacam-se Python e JavaScript, cada uma com características específicas e propósitos distintos.

Python é uma linguagem de alto nível conhecida por sua sintaxe clara e objetiva, o que facilita seu aprendizado e aplicação em diversas áreas da computação. É amplamente utilizada para automação de processos, análise de dados, inteligência artificial e desenvolvimento web. Uma das principais vantagens do Python está em sua vasta biblioteca de pacotes, que simplificam tarefas complexas como leitura de arquivos, manipulação de dados e conexão com bancos de dados.

JavaScript, por sua vez, é uma linguagem essencial para o desenvolvimento frontend de aplicações web, sendo responsável por tornar as interfaces interativas e dinâmicas. Com o tempo, surgiram frameworks como o React.js, que permite a criação de componentes reutilizáveis, tornando o desenvolvimento mais organizado e eficiente. React é bastante adotado na construção de interfaces modernas e responsivas, promovendo uma experiência fluida ao usuário.

Além das linguagens, bibliotecas específicas são frequentemente empregadas para executar tarefas específicas. Por exemplo, a biblioteca panda, desenvolvida em Python, permite a manipulação de dados em estruturas tabulares, sendo muito utilizada em aplicações que envolvem transformação e limpeza de. Para a leitura de arquivos PDF, ferramentas como PyPDF2 e pdfplumber são comumente utilizadas, enquanto bibliotecas como openpyxl permitem interações com planilhas Excel, como leitura, escrita e atualização de células (VanderPlas, 2016).

Para o desenvolvimento de APIs modernas, frameworks como o FastAPI têm se destacado. Escrito em Python, ele permite a criação de interfaces de comunicação entre sistemas de forma rápida, segura e com excelente desempenho. Ele utiliza conceitos de programação assíncrona e tipagem estática, promovendo maior clareza no código e geração automática de documentação (FastAPI, 2024).

O armazenamento e a organização dos dados são geralmente gerenciados por Sistemas de Gerenciamento de Banco de Dados (SGBDs). Entre os relacionais, o PostgreSQL se destaca por sua robustez, suporte a transações complexas, integridade referencial e segurança. Bancos relacionais organizam os dados em tabelas interligadas, permitindo consultas estruturadas por meio da linguagem SQL.

Para facilitar o processo de desenvolvimento, são utilizados ambientes integrados de desenvolvimento (IDEs), como o Visual Studio Code (VSCode). Esse editor de código moderno é compatível com diversas linguagens, possui suporte a extensões e ferramentas de depuração, e é amplamente adotado por desenvolvedores pela sua leveza e praticidade. Outros recursos como controle de versões (Git), terminais integrados e plugins tornam o VSCode uma plataforma versátil tanto para iniciantes quanto para profissionais experientes.

A combinação dessas tecnologias possibilita o desenvolvimento de sistemas completos e robustos, com backend eficiente, frontend responsivo e integração facilitada com bases de dados e outras aplicações.