

**Caio Gabriel da Silva Pimenta Cardeliquio**

**Gabriel Gustavo Permegiani Godoy**

**Henrique Criado Roldan**

**Julio Cesar Barbosa Ribeiro**

**Olivio Marques Brito Neto**

**ESTAGIOTECH**

**Projeto Integrador**

Disciplinas Envolvidas: Engenharia de Software, Tópicos Avançado de Banco de Dados Relacional e não Relacional, Técnicas Avançadas de Programação, Técnicas Avançadas de Programação Web e Mobile, Gestão Ágil de Projetos de Software.

Jales

2023

SUMÁRIO

[1 INTRODUÇÃO 3](#_Toc144809989)

[2 LEVANTAMENTO DE REQUISITOS DE SOFTWARE 4](#_Toc144809990)

[2.1 Descrição dos objetivos do sistema 4](#_Toc144809991)

[2.2 Descrição do sistema atual 4](#_Toc144809992)

[2.3 Descrição dos principais problemas 4](#_Toc144809993)

[2.4 Descrição dos requisitos funcionais 4](#_Toc144809994)

[2.5 Descrição dos requisitos não funcionais 4](#_Toc144809995)

[3 VISÃO DE CASO DE USO – UML 5](#_Toc144809996)

[3.1 Diagrama de Classes 5](#_Toc144809997)

[3.2 Dicionário de classes 5](#_Toc144809998)

[3.3 Definição dos Atores 5](#_Toc144809999)

[3.4 Lista de Casos de Uso 5](#_Toc144810000)

[3.4. Diagrama de Casos de Uso 5](#_Toc144810001)

[3.5. Diagrama de Casos de uso individuais 6](#_Toc144810002)

[3.6. Diagrama de Sequência 7](#_Toc144810003)

[3.7. Diagrama de Comunicação 7](#_Toc144810004)

[3.8. Diagrama de Atividade 7](#_Toc144810005)

[4 DEFINIÇÃO DA INTERFACE COM O USUÁRIO (UX) (3º semstre) 8](#_Toc144810006)

[4.1 Descrição de cenário 8](#_Toc144810007)

[4.2 Descrição de personas 8](#_Toc144810008)

[4.3 Protótipos de tela 8](#_Toc144810010)

[5 BANCO DE DADOS 9](#_Toc144810011)

[5.1 Modelo Entidade Relacionamento 9](#_Toc144810012)

[5.2 Script das tabelas 9](#_Toc144810013)

[6 ARQUITETURA DE SOFTWARE 10](#_Toc144810014)

[6.1 Arquitetura de desenvolvimento 10](#_Toc144810015)

[6.2 Telas do sistema 10](#_Toc144810016)

[7 CONCLUSÃO 11](#_Toc144810017)

[8 REFERÊNCIAS 12](#_Toc144810018)

# INTRODUÇÃO

Este Projeto Integrador tem como objetivo facilitar a parte burocrática de um processo de contratação através de um sistema informatizado, de fácil manuseio e de interface limpa. Para o desenvolvimento deste projeto as ferramentas utilizadas serão semelhantes ou iguais as que o grupo utiliza na graduação, para melhor entendimento dos integrantes e contando com o auxílio dos docentes da instituição, caso ocorra o uso inadequado da plataforma ou quem sabe alguma dúvida em relação com alguma ferramenta.

No cenário de desenvolvimento deste software a automação dos processos se faz necessária, pois as etapas atuais estão relativamente ultrapassadas e lentas de certa forma. Isso acaba atrasando a inserção dos estudantes no mercado de trabalho e de certa forma, desmotivando os mesmos de continuar na instituição, por não verem oportunidades para o seu ingresso no mercado de trabalho.

O sistema tem como principal função a homologação de documentos, além do objetivo de facilitar e agilizar o processo de contratação. Neste sistema será encontrado tudo o que diz respeito ao termo compromisso de estágio entre aluno e instituição concedente, todos os documentos. Estes documentos são por exemplo os documentos pessoas do contratado, o contrato entre as partes com todos os detalhes incluindo: Horas semanais, remuneração, supervisor responsável, responsável legal. Outros documentos também estarão presentes como por exemplo os relatórios parciais e finais feitos pelo próprio estagiário, neste documento será descrito tudo o que ele fez enquanto desenvolvia as suas atividades, ele será divido em duas partes, primeira parte com entrega na metade do contrato e a outra metade no final do contrato.

# LEVANTAMENTO DE REQUISITOS DE SOFTWARE

## Descrição dos objetivos do sistema

Este projeto tem como objetivo desburocratizar o processo de contratação dos alunos da Fatec, auxiliando não somente eles como também os profissionais da instituição. Proporcionando uma ferramenta de fácil acesso e entendimento da comunidade, pode-se facilitar este processo, principalmente com a relação empresa e instituição

## Descrição do sistema atual

O sistema atualmente não é informatizado, porém existe um processo: Para recrutar alunos é por recomendação do professor ou por grupo através do WhatsApp; Caso o aluno seja um dos escolhidos ele receberá uma ligação afim de ficar por dentro sobre o dia da entrevista, os documentos necessários; Se este for contratado receberá outra ligação recebendo as informações contratuais.

Percebemos que em relação a organização dos processos a instituição tem um norte muito bem definido, pois já se tem as pessoas responsáveis, o que falta mesmo é algo mais automatizado, retirando processos antigos e inovando os mesmos.

## Descrição dos principais problemas

Com base nos problemas relatado por funcionários da própria instituição, as maiores dificuldades relatadas é a falta de organização das contratações e falta de acompanhamento durante as etapas de contratação, para um melhor feedback sobre como está a contratação.

## Descrição dos requisitos funcionais

O sistema deve fornecer ao estagiário as etapas de contratação e toda a parte burocrática: seus documentos, relatórios e tarefas.

Já para os administradores, o software fornecerá todas as informações possíveis dentro do sistema.

Para os supervisores de estágio e concedentes do estágio aparecerão as informações de acordo com os estagiários que estes serão responsáveis, porém com autorização de editar e validar se uma tarefa foi realmente concluída.

Além dessas funções o sistema irá: cadastrar, editar, excluir, alterar e cadastra dados, estas funcionalidades vão funcionar nas seguintes classes do sistema: Supervisor Estágio, Estagiário, Instituição Ensino, Concedente Estágio e Administrador.

## Descrição dos requisitos não funcionais

Os requisitos não funcionais presentes no sistema serão: segurança, desempenho, autenticação e autorização, usabilidade, interface agradável, desempenho de banco de dados e backup realizado quinzenalmente.

# VISÃO DE CASO DE USO – UML

## 3.1 Diagrama de Classes

*Fazer um texto explicativo sobre o diagrama e inserir abaixo. As imagens devem conter o nome da figura e abaixo a fonte*

*Colocar em outra cor as classes que foram implementadas nas aulas de programação*

## 3.2 Dicionário de classes

Aluno: tem como objetivo definir quem é um possível candidato a vagas e novo estagiário da instituição.

**Quadro 1** — Descrição Classe Aluno

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| idAluno | Integer | Utilizado para localizar o aluno dentro do sistema |
| NomeAluno | String | Nomeia os usuários do sistema |
| Idade | Integer | Informa a idade dos usuários |
| RG | String | Registro geral dos usuários |
| StatusAluno | Boolean | Determina e informa qual o cargo do aluno a qual etapa esta da contratação |
| NumeroMatricula | String | Ordena e numera as matrículas |
| AreaInteresse | String | Determina qual área o estagiário tem interesse em atuar |
| Habilidades | String | É informado quais são as habilidades que o usuário pode oferecer |
| Experiencias | String | Informa quais foram as experiencia que o usuário possui |
| DisponibilidadesHorários | String | Quais os horários o usuário tem disponível |
| Currículo | String | É apresentado o currículo do usuário |
| CPF | String | É o cadastro de pessoa física |
| DataNacimento | dateTime | É informado a data de nascimento do usuário |
| NivelEscolaridade | String | Informa qual o nível de escolaridade do usuário |
| Telefone | String | Mostra o telefone do usuário |
| E-mail | String | E-mail do usuário |
| Endereço | String | Informa o endereço do aluno |
| Gênero | String | Especifica o gênero do usuário |
| Bairro | String | Informa o bairro do usuário |
| Cep | String | É informado o código de endereçamento postal |

Fonte: Elaborado pelos autores

Vagas: tem como objetivo definir qual a vaga está disponível para o candidato e todas as suas exigências.

**Quadro 2** — Descrição Classe Vagas

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| idVagas | Integer | É utilizado para localizar dentro dos sistemas cada uma das vagas |
| IdConcendente | Integer | É usado para informar qual o identificador da empresa que está contratando o estagiário |
| QuantidadeVagas | Integer | Informa exatamente o número de vagas existentes de determinada empresa |
| DataPublicaçao | Date | Informa e oficializa a postagem da vaga |
| DataLimite | dateTime | É defina uma data limite para a inscrição do estagiário para determinada vaga |
| Localidade | String | Informa a localidade em que a empresa se encontra rua, bairro e cidade |
| Descriçao | String | Na descrição será descrito todas as informações referentes a vaga de estágio |
| Titulo | String | Informa como um previa do que o estagiário irá exercer na vaga |
| LocaldeTrabalho | String | Local de trabalho se refere a empresa ou local físico que o estagiário irá trabalhar |
| HorarioEntrada | dateTime | Ira definir a entrada em que o estagiário deverá estar para cumprir sua carga horária |
| horarioSaida | dateTime | Hora de saída de acordo com a carga horaria presente no contrato e as horas que precisam ser cumpridas na semana |
| totalHorasSemanais | String | Define a carga horaria total estipulada pelo contrato de acordo com as regras de estágio |

Fonte: Elaborado pelos autores

Contrato Estágio: tem como objetivo definir quis os termos e reponsabilidades irão conter no contrato do aluno.

**Quadro 3** — Descrição Classe Contrato Estágio

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| idContratoEstagio | Integer | Utilizado para identificar os contratos presentes dentro do sistema |
| StatusContratoEstagio | Boolean | Mostra em qual etapa está a contratação do usuário |
| NotaFinal | Double | Nota que avalia o desempenho do estagiário |
| Situação | Integer | Mostra qual a atual situação do estagiário |
| HorárioEntrada | Integer | De acordo com o período e a carga horaria é definido o horário de entrada |
| HorarioSaida | Integer | De acordo com o horário de entrada é definido o horário de saída |
| DataInicio | Integer | É definido no contrato o prazo de início contrato como estagiário |
| DataFim | Integer | De acordo com as regras de estágio é definido um prazo limite de dois anos ou menos para o estagiário |
| Salario | Integer | Salário pode variar de acordo com a empresa e de acordo a área em que ele irá atuar |
| CargaSemanal | Integer | É definido quanto de horas o estagiário deverá cumprir na semana |
| CargaTotal | Integer | É somado a carga horaria total que o estagiário deverá entregar |

Fonte: Elaborado pelos autores

Usuário: tem como objetivo definir as informações de cadastro e login do usuário

**Quadro 4** — Descrição Classe Usuário

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| loginUsuario | String | É o que autêntica o usuário no sistema |
| senhaUsuario | String | É o utilizado juntamente com o login para a autenticação do usuário |
| statusUsuario | Boolean | Mostra qual o status do usuário dentro dos sistemas |
| emailUsuario | Integer | É um meio de comunicação com o usuário |

Fonte: Elaborado pelos autores

Curso: tem como objetivo definir as informações do curso que o aluno está matriculado

**Quadro 5** — Descrição Classe Curso

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| IdCurso | Integer | Utilizado para identificar o curso do usuário |
| nomeCurso | Integer | Nomeia e identifica o curso |

Fonte: Elaborado pelos autores

Matrícula: tem como objetivo definir as informações da matrícula realizada pelo aluno

**Quadro 6** — Descrição Classe Matrícula

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| idMatricula | Integer | Identifica as matrículas que foram feitas |
| registroAcademico | Integer | Mostra o registro acadêmico do aluno |

Fonte: Elaborado pelos autores

Instituição Ensino: tem como objetivo definir as informações da instituição de ensino que pode ou não conceder uma vaga ou estágio

**Quadro 7** — Descrição Classe Instituição Ensino

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |

Fonte: Elaborado pelos autores

Candidato: tem como objetivo definir as informações do candidato, nessa classe ele passa de um aluno para postulante a vaga de estágio

**Quadro 8** — Descrição Classe Candidato

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| idCandidato | Integer | Identifica o candidato |
| dataInscricao | Date | Refere-se ao dia em que o aluno se inscreveu para aquela vaga |
| Situacao | String | Mostra a situação do candidato perante a vaga |

Fonte: Elaborado pelos autores

Processo: tem como objetivo de mostrar ao candidato como está ocorrendo o processo seletivo referente a sua vaga

**Quadro 9** — Descrição Classe Processo Seletivo

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| idProcessoSeletivo | Integer | Identifica qual o processo seletivo |
| dataCandidatura | Date | Refere-se ao dia em que o processo seletivo foi publicado |

Fonte: Elaborado pelos autores

Coordenador Estágio: tem como objetivo identificar e registrar um dos atores do sistema

**Quadro 10** — Descrição Classe Coordenador Estágio

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| idCoordenadorEstagio | Integer | Identifica quem é o coordenador de estágio |
| dataCadastro | Date | Refere-se ao dia em que o coordenador de estágio foi cadastrado no sistema |
| StatusCoordenadorEstagio | Boolean | Define o status do coordenador, se ele está inativo ou ativo no banco |

Fonte: Elaborado pelos autores

Concedente: tem como objetivo identificar qual a empresa que vai conceder a vaga e o estágio para um candidato

**Quadro 11** — Descrição Classe Concedente

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| idConcedente | Integer | Identifica qual é a empresa que irá conceder o estágio |
| ResposnsavelEstagioConcedente | String | Mostra quem é a pessoa de dentro da empresa que é responsável por conceder o estágio |

Fonte: Elaborado pelos autores

Cargo: tem como objetivo mostrar qual o cargo os novos estagiários irão trabalhar

**Quadro 12** — Descrição Classe Cargo

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| idCargo | Integer | Identifica qual é o cargo |
| descricaoCargo | String | Descreve como será o cargo exercido com detalhes |
| tipoCargo | String | Define quais funções o estagiário irá exercer na instituição |

Fonte: Elaborado pelos autores

Requisito Vagas: mostra quais são os requisitos para que um aluno entre naquela vaga específica

**Quadro 13** — Descrição Classe Requisito Vagas

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| idRequisitoVagas | Integer | Identifica qual é o requisito daquela vaga |
| requisitosVagas | String | Detalha quais os requisitos específicos para aquela vaga |

Fonte: Elaborado pelos autores

Supervisor Estágio: tem como objetivo identificar e registrar um dos atores do sistema

**Quadro 14** — Descrição Classe Requisito Vagas

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| idSupervisor | Integer | Identifica qual é o identificador do supervisor de estágio |
| statusSupervisor | Boolean | Define o status do supervisor, se ele está inativo ou ativo no banco |

Fonte: Elaborado pelos autores

Documento: tem como objetivo mostrar qual o documento que está sendo cadastrado

**Quadro 15** — Descrição Classe Requisito Vagas

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| idDocumeto | Integer | Identifica qual é o identificador do documento |
| descricaoDocumento | String | Descreve como está aquele documento no momento |
| situacaoDocumento | String | Define qual a situação que o documento se encontra |

Fonte: Elaborado pelos autores

Apontamento: tem como objetivo mostrar qualquer observação referente a um documento específico

**Quadro 16** — Descrição Classe Apontamento

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| idApontamento | Integer | Identifica qual é o apontamento |
| descricaoApontamento | String | Descreve o motivo daquele apontamento estar acontecendo |
| dataApontamento | Date | Define qual a data que aquele apontamento foi feito |

Fonte: Elaborado pelos autores

Tipo Documento: tem como objetivo mostrar qual o tipo do documento

**Quadro 17** — Descrição Classe Tipo Documento

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| idTipoDocumento | Integer | Identifica qual é o tipo do documento |
| descricaoTipoDocumento | String | Mostra qual a descrição do tipo documento |

Fonte: Elaborado pelos autores

Documento Versão: tem como objetivo mostrar qual é a versão de um determinado documento

**Quadro 18** — Descrição Classe Documento Versão

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| idDocumentoVersao | Integer | Identifica qual é a versão do documento |
| comentarioDocumentoVersao | String | Mostra os comentários feitos na versão atual do documento |
| anexoDocumentoVersao | String | Identifica qual o anexo da versão do documento |
| dataDocumentoVersao | Date | Mostra a data da versão do documento |
| situacaoDocumentoVersao | String | Identifica a situação da versão do documento |

Fonte: Elaborado pelos autores

Tipo Estágio: tem como objetivo mostrar qual é o tipo do estágio

**Quadro 19** — Descrição Classe Tipo Estágio

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |
| idTipoEstagio | Integer | Identifica qual o tipo do estágio |
| descricaoTipoEstagio | String | Descreve como será aquele estágio |

Fonte: Elaborado pelos autores

Documento Necessário: É uma classe gerada partindo de tipo estágio e tipo documento, ela identifica qual o documento o aluno terá que apresentar para provar que já faz algum estágio ou não

**Quadro 20** — Descrição Classe Documento Necessário

|  |  |  |
| --- | --- | --- |
| Atributo | Tipo | Descrição |

Fonte: Elaborado pelos autores

## 3.3 Definição dos Atores

*Inserir a figura dos atores e abaixo fazer uma descrição sobre o ator (es).*

## 3.4 Lista de Casos de Uso

*(Descrever sobre a Lista de Casos de Uso)*

Quadro 1 – Lista de Casos de Uso

| **Nº** | **Descrição do Caso de Uso** | **Entrada** | **Caso de Uso** | **Resposta** |
| --- | --- | --- | --- | --- |
| 01 | Funcionário Cadastra Livro | dados livro | Cadastrar Livro | Msg01 |
| 02 | Funcionário Cadastra Autor | dados autor | Cadastrar Autor | Msg02 |

## 3.4. Diagrama de Casos de Uso

*Fazer um texto explicativo sobre o diagrama e inserir-lo abaixo. As imagens deve conter o nome da figura e abaixo a fonte.*

**Figura 01** — Diagrama de Contexto – Visão do Paciente.



Fonte: Silva, 2016.

## 3.5. Diagrama de Casos de uso individuais

*(2 principais: cadastro, consulta, relatório, agendamento, controle, etc..)*

*Inserir o diagrama e abaixo colocar a documentação do mesmo.*

3.5.1 – Caso de uso: Cadastrar Atendimento

**Figura 02** — Diagrama de Contexto – Visão do Paciente

**

Fonte: Silva, 2016.

**Fluxo Normal**

1-AtorAdm solicita exclusão do registro da pessoa

2-Sistema envia mensagem 06: "Deseja excluir o registro?"

3-AtorAdm confirma exclusão do registro

4-Sistema exclui registro

5-Sistema envia msg07: "Registro excluído com sucesso!"

6-Sistema atualiza banco de dados e retorna ao Caso de Uso "Alterar Pessoa"

**Fluxo Exceção**

2-Sistema envia mensagem 06: "Deseja excluir o registro?"

2.1-AtorAdm não confirma exclusão do registro

2.2-Sistema cancela ação

2.3-Sistema retorna ao caso de Uso "Alterar Pessoa"

Fluxo Alternativo

4-Sistema exclui registro

4.1-Sistema identifica erros de conexão com banco de dados ao excluir registro

4.2-Sistema envia msg07:"Erro s de conexão de BD"

4.3-Sistema retorna ao item 1

## 3.6. Diagrama de Sequência

*(2 principais: cadastro, consulta, relatório, agendamento, controle, etc..)*

*Inserir o diagrama e abaixo colocar a documentação do mesmo.*

## 3.7. Diagrama de Comunicação

*(1 diagrama relevante do projeto)*

## 3.8. Diagrama de Atividade

*(1 diagrama de uma ação específica do sistema)*

# DEFINIÇÃO DA INTERFACE COM O USUÁRIO (UX) (3º semstre)

## Descrição de cenário

Conforme escrito no livro "O Guia para Projetar UX" de Russ Unger e Carolyn Chandler, os cenários representam narrativas detalhadas que ilustram as interações entre um usuário e um sistema. Estas histórias oferecem uma compreensão abrangente ao contextualizar a persona, seus objetivos, as ações específicas em busca desses objetivos, os desafios enfrentados e os desfechos das interações. Os cenários de persona são ferramentas fundamentais no direcionamento do design de experiência do usuário, permitindo uma avaliação minuciosa da eficácia do sistema em diferentes circunstâncias.

Figura ?? – Imagem do primeiro cenário

Fonte: Elaborado pelos autores

Figura ?? – Imagem do segundo cenário



Fonte: Elaborado pelos autores

## Descrição de personas

De acordo com "O Guia para Projetar UX" de Russ Unger e Carolyn Chandler, personas constituem representações fictícias de usuários envolvidos na interação com um sistema específico. Seu desenvolvimento é direcionado à compreensão aprofundada das necessidades, comportamentos e motivações desses usuários. A criação de personas visa proporcionar aos designers e desenvolvedores uma visão mais humanizada e centrada no usuário, possibilitando a concepção de soluções que correspondam de maneira eficaz às expectativas e requisitos dos usuários finais.

Figura ?? – Imagem da primeira persona



Fonte: Elaborado pelos autores

Figura ?? – Imagem da segunda persona



Fonte: Elaborado pelos autores

## Protótipos de tela

Como afirma Don Norman, autor de "The Design of Everyday Things", "protótipos são uma ferramenta fundamental para explorar possíveis soluções de design antes da implementação completa". Os protótipos de tela, são representações tangíveis e interativas da proposta de design, permitindo a exploração, comunicação e refinamento de ideias". No contexto do ESTAGIOTECH, esses esboços desempenham o papel de uma primeira ideia visual de como será a interface do usuário.

Assim, os modelos iniciais de tela não só promovem a comunicação efetiva entre as equipes de design e desenvolvimento, como também têm um papel crucial na detecção antecipada de adaptações necessárias, resultando em economia de tempo e recursos.

Figura ?? - Imagem do Prototipo da Tela de Cadastro do Administrador

Fonte: Elaborado pelos autores

O protótipo de tela para o Cadastro do Administrador no ESTAGIOTECH é o processo de cadastro visual do administrador do sistema. O layout podemos ver os campos essenciais para realizar o cadastro, dentre eles, o E-mail, senha a confirmação de senha e o CPF do administrador, além dos campos de aceitar os termos de uso. O botão “Já tenho cadastro” é a opção caso o cadastro já tenha sido feito.

Figura ?? - Imagem do Prototipo da Tela de Login do Administrador

Fonte: Elaborado pelos autores

O protótipo de tela para o Login do Administrador no ESTAGIOTECH é o processo que a representação visual do processo de autenticação do administrador. No layout temos os campos essenciais para realizar o login, que são o E-mail do administrador e uma senha previamente cadastrados, com uma abordagem minimalista e de fácil compreensão. A inclusão de elementos como botões de "Entrar" caso o administrador já tenha cadastro, “Criar conta” que o redireciona para a figura anterior, "Esqueceu a Senha" se por um acaso ele tenha esquecido sua senha.

Figura ?? - Imagem do Prototipo da Tela de Controle dos Estagiários

Fonte: Elaborado pelos autores

O protótipo de tela para o Controle de Estagiários é responsável por mostrar ao administrador todos os estagiários que estão cadastrados no sistema, afim do controle ser ainda perante a função que o estagiário está desempenhando na instituição ou na empresa concedente. No layout podemos ver o nome do estagiário, um breve resumo sobre a sua função e partes burocráticas contratuais, como sua carga horária, seu salário quando ele começou e quando terminará o seu estágio. Além de uma função de status, no sistema não pretendemos excluir o estagiário e sim desativar ele, para ter um controle ainda maior sobre quem já estagiou pela instituição ou por outra empresa.

Figura ??- Imagem do Prototipo da Tela Painel de Tarefas do Estagiário para o Administrador



Fonte: Elaborado pelos autores

O protótipo da tela Painel de Tarefas é responsável por mostrar ao administrador do sistema como estão as tarefas desempenhadas pelo estagiário, temos quais as tarefas o aluno deverá desempenhar, o dia que foi atribuída e o dia que ele terá que entregar essa tarefa, tendo também a barra de progresso, onde vemos como vai o avanço do aluno de acordo com o cumprimento da atividade, essa atividade só será finalizada caso alguém superior aprove e fiscalize o real cumprimento da tarefa. Temos as abas onde mostramos o contrato do aluno e as tarefas concluídas e aprovadas do aluno, além das informações rápidas sobre o contrato como mostrado também na figura anterior.

Figura ?? - Imagem do Prototipo da Tela de Tarefas Concluidas para o Administrador



Fonte: Elaborado pelos autores

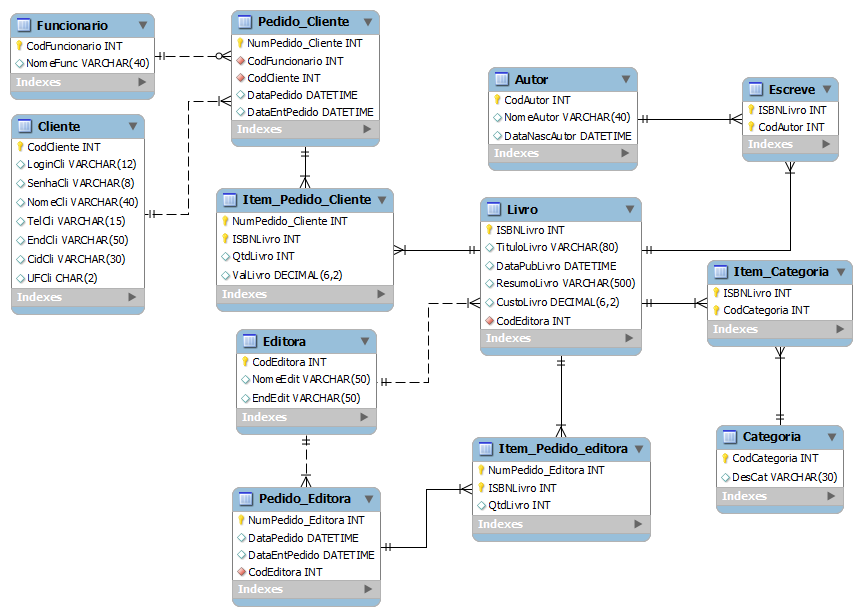
O protótipo da tela de Tarefas Concluídos tem o objetivo de mostrar ao administrador como estão as tarefas concluídas do estagiário, tendo qual a tarefa o aluno desempenhou quando foi finalizada o status de aprovado juntamente com a barra de progresso em 100%, reforçando que a tarefa foi realmente finalizada, contendo também as informações básicas contratuais do aluno especifico.

# BANCO DE DADOS

## Modelo Entidade Relacionamento

O Modelo Entidade-Relacionamento (MER), segundo o livro “Banco de dados: projeto e implementação”, é uma representação gráfica das entidades (que são representações de objetos do mundo real), dos atributos (as características de uma entidade) e dos vínculos (as conexões entre diferentes entidades) que compõem o sistema de armazenamento de dados. É uma ferramenta que ajuda a capturar interações intricadas e a organizar os dados de forma clara e compreensível.

Figura ?? – Mapeamento do Objeto Relacional



Fonte: Elaborado pelos autores

## Script das tabelas

Script das tabelas, conforme abordado no livro “Banco de dados: Projeto e Implementação”, consistem em diretrizes ou conjuntos de comandos redigidos numa linguagem de consulta estruturada (SQL) com o propósito de criar, modificar ou remover tabelas em um banco de dados. Esses conjuntos de instruções são empregados para estabelecer a estrutura de um banco de dados relacional, abrangendo definições de tabelas, campos e restrições.

Quadro ?? – Script SQL – Tabela Aluno

|  |
| --- |
| CREATE TABLE aluno (  alunoid integer GENERATED BY DEFAULT AS IDENTITY,  nome character varying(150) NOT NULL,  idade integer NOT NULL,  rg character varying(12) NOT NULL,  statusaluno boolean NOT NULL,  numeromatricula character varying(50) NOT NULL,  areainteresse character varying(100) NOT NULL,  habilidades character varying(100) NOT NULL,  experiencias character varying(350) NOT NULL,  disponibilidadehorario character varying(35) NOT NULL,  curriculo text NOT NULL,  cpf character varying(14) NOT NULL,  cidade character varying(50) NOT NULL,  datanascimento timestamp with time zone NOT NULL,  nivelescolaridade character varying(80) NOT NULL,  telefone character varying(14) NOT NULL,  email character varying(50) NOT NULL,  endereco character varying(50) NOT NULL,  genero character varying(30) NOT NULL,  bairro character varying(30) NOT NULL,  cep character varying(9) NOT NULL,  CONSTRAINT "PK\_aluno" PRIMARY KEY (alunoid)  ); |

Fonte: Elaborado pelos autores.

Quadro ?? – Script SQL – Tabela Concedente

|  |
| --- |
| CREATE TABLE concedente (  concedenteid integer GENERATED BY DEFAULT AS IDENTITY,  razaosocial character varying(80) NOT NULL,  responsavelestagio character varying(50) NOT NULL,  cnpj character varying(16) NOT NULL,  localidade character varying(50) NOT NULL,  CONSTRAINT "PK\_concedente" PRIMARY KEY (concedenteid)  ); |

Fonte: Elaborado pelos autores.

Quadro ?? – Script SQL – Tabela Curso

|  |
| --- |
| CREATE TABLE "Curso" (  "idCurso" integer GENERATED BY DEFAULT AS IDENTITY,  "nomeCurso" character varying(200) NOT NULL,  CONSTRAINT "PK\_Curso" PRIMARY KEY ("idCurso")  ); |

Fonte: Elaborado pelos autores.

Quadro ?? – Script SQL – Tabela Documento

|  |
| --- |
| CREATE TABLE documento (  documentoid integer GENERATED BY DEFAULT AS IDENTITY,  descricao character varying(200) NOT NULL,  situacao character varying(200) NOT NULL,  CONSTRAINT "PK\_documento" PRIMARY KEY (documentoid)  ); |

Fonte: Elaborado pelos autores.

Quadro ?? – Script SQL – Tabela TipoDocumento

|  |
| --- |
| CREATE TABLE tipodocumento (  tipodocumentoid integer GENERATED BY DEFAULT AS IDENTITY,  descricao character varying(200) NOT NULL,  CONSTRAINT "PK\_tipodocumento" PRIMARY KEY (tipodocumentoid)  ); |

Fonte: Elaborado pelos autores.

Quadro ?? – Script SQL – Tabela TipoEstagio

|  |
| --- |
| CREATE TABLE tipoestagio (  tipoestagioid integer GENERATED BY DEFAULT AS IDENTITY,  descricao character varying(200) NOT NULL,  CONSTRAINT "PK\_tipoestagio" PRIMARY KEY (tipoestagioid)  ); |

Fonte: Elaborado pelos autores.

Quadro ?? – Script SQL – Tabela Vagas

|  |
| --- |
| CREATE TABLE vagas (  vagasid integer GENERATED BY DEFAULT AS IDENTITY,  quantidade integer NOT NULL,  datapublicacao timestamp with time zone NOT NULL,  datalimite timestamp with time zone NOT NULL,  localidade character varying(80) NOT NULL,  descricao character varying(200) NOT NULL,  titulo character varying(80) NOT NULL,  localidadetrabalho character varying(20) NOT NULL,  horarioentrada character varying(20) NOT NULL,  horariosaida character varying(20) NOT NULL,  totalhorassemanais character varying(20) NOT NULL,  "concedenteId" integer NOT NULL,  CONSTRAINT "PK\_vagas" PRIMARY KEY (vagasid),  CONSTRAINT "FK\_vagas\_concedente\_concedenteId" FOREIGN KEY ("concedenteId") REFERENCES concedente (concedenteid) ON DELETE CASCADE  ); |

Fonte: Elaborado pelos autores.

Quadro ?? – Script SQL – Tabela DocumentoVersao

|  |
| --- |
| CREATE TABLE documentoversao (  documentoversaoid integer GENERATED BY DEFAULT AS IDENTITY,  comentario character varying(200) NOT NULL,  anexo character varying(200) NOT NULL,  data character varying(200) NOT NULL,  situacao character varying(200) NOT NULL,  documentoid integer NOT NULL,  CONSTRAINT "PK\_documentoversao" PRIMARY KEY (documentoversaoid),  CONSTRAINT "FK\_documentoversao\_documento\_documentoid" FOREIGN KEY (documentoid) REFERENCES documento (documentoid) ON DELETE CASCADE  ); |

Fonte: Elaborado pelos autores.

Quadro ?? – Script SQL – Tabela Cargo

|  |
| --- |
| CREATE TABLE cargo (  cargoid integer GENERATED BY DEFAULT AS IDENTITY,  descricao character varying(200) NOT NULL,  tipo character varying(50) NOT NULL,  vagasid integer NOT NULL,  CONSTRAINT "PK\_cargo" PRIMARY KEY (cargoid),  CONSTRAINT "FK\_cargo\_vagas\_vagasid" FOREIGN KEY (vagasid) REFERENCES vagas (vagasid) ON DELETE CASCADE  ); |

Fonte: Elaborado pelos autores.

# ARQUITETURA DE SOFTWARE

## 6.1 Arquitetura de desenvolvimento

A arquitetura de software, constitui a estrutura essencial que orienta o desenvolvimento da aplicação, traduzindo requisitos do sistema em uma representação tangível. No contexto do ESATGIOTECH, a estrutura de software constitui o alicerce essencial do sistema, oferecendo uma perspectiva estruturada e unificada sobre a interação dos elementos entre si. Como dito por Ralph Johnson, um dos autores do livro "Design Patterns: Elements of Reusable Object-Oriented Software", a arquitetura de software atua como a "espinha dorsal" que mantém a aplicação, delineando as conexões entre os distintos módulos e assegurando uma estrutura sólida e resistente.

A necessidade de uma arquitetura sólida no ESTAGIOTECH vai além da mera organização do código. Conforme dito por Martin Fowler, especialista em arquitetura de software, "a relevância da arquitetura é evidente ao lidar com a complexidade do desenvolvimento de software. ". No âmbito do ESTAGIOTECH, a complexidade do sistema requer uma arquitetura meticulosamente estruturada, não só para simplificar a implementação presente, mas também para viabilizar a manutenção subsequente e a expansão contínua do projeto.

Portanto, a arquitetura de software no ESTAGIOTECH transcende a formalidade técnica, sendo uma estratégia essencial para garantir a clareza na concepção, a flexibilidade na evolução e a capacidade de resposta às necessidades em constante mudança.

## 6.1.1 Linguagem de Programação C#

Figura ?? – Imagem da Logo do C#



Fonte: Cwork Sistemas(2018)

Acima podemos ver a logo da linguagem implementada no ESTAGIOTECH, o C# desempenha um papel crucial na construção de uma API robusta e eficiente. Como destacado por Anders Hejlsberg, arquiteto-chefe da linguagem C# na Microsoft, "C# é projetado para ser uma linguagem de programação simples, moderna e orientada a objetos". A escolha dessa linguagem casa perfeitamente com a necessidade do projeto e com o que foi aprendido no curso, com o auxílio do docente capacitado, fica melhor o suporte para o grupo com alguma dúvida, tendo também como uma das principais características a fácil manutenção.

Martin Fowler, especialista em padrões de arquitetura de software, ressalta que "C# é particularmente adequado para a implementação de arquiteturas em camadas". Ao utilizar essas camadas dentro do ESTAGIOTECH temos como vantagem coesão e flexibilidade.

Além disso, Scott Guthrie, Vice-Presidente Executivo da Microsoft Cloud + AI, destaca a integração do C# com o ecossistema .NET, por oferecer acesso a uma extensa variedade de bibliotecas e ferramentas, essa integração eficiente se mostra crucial para assegurar uma comunicação fluida entre as diversas camadas da aplicação e o banco de dados do ESTAGIOTECH. Esse processo cria uma base sólida para o desenvolvimento contínuo e sustentável da API.

## 6.1.2 Visual Studio

Figura ?? – Imagem da Logo do Visual Studio



Fonte: 1000 logos(2023)

Acima vemos a logo do Visual Studio que foi escolhido como ambiente de desenvolvimento para a implementação da API em C# deste, respaldada por sua reputação consolidada e recursos abrangentes. Como enfatiza Anders Hejlsberg, arquiteto-chefe da linguagem C# na Microsoft, o Visual Studio é reconhecido por sua robustez e conjunto de ferramentas integradas, proporcionando um ambiente de desenvolvimento completo para a criação de aplicativos empresariais de alta qualidade, oferendo funcionalidades avançadas para a implementação da arquitetura em camadas, deixando melhor a implementação e entendimento do código fonte, e isso é fundamental para as necessidades do ESTAGIOTECH.

A interligação entre o Visual Studio e o ecossistema .NET é ressaltada por especialistas como Scott Guthrie, Vice-Presidente Executivo da Microsoft Cloud + AI. Essa integração simplifica significativamente o processo de desenvolvimento em C#, facilitando uma comunicação contínua entre as variadas camadas da aplicação e o banco de dados do projeto.

Assim, ao escolher o Visual Studio como o ambiente principal para o desenvolvimento da API em C#, a equipe do ESTAGIOTECH almeja não apenas uma ferramenta robusta e estabelecida, mas também um ambiente que simplifique a adoção de boas práticas de arquitetura e promova um fluxo de trabalho eficiente para a concepção de uma aplicação escalável e de excelência em qualidade.

## 6.1.3 Visual Studio Code

Figura ?? – Imagem da Logo do Visual Studio



Fonte: Medium(2019)

A escolha do Visual Studio Code como ambiente de desenvolvimento para o ESTAGIOTECH reflete a preferência por uma ferramenta versátil e amplamente adotada na comunidade de desenvolvimento. Martin Fowler, um autor de renome, ressalta que o Visual Studio Code é reconhecido por sua leveza, simplicidade e capacidade de expansão, fornecendo um ambiente de codificação eficaz e focado no desenvolvedor. Com uma interface intuitiva e suporte sólido para múltiplas linguagens de programação, o Visual Studio Code se destaca como uma escolha versátil.

Além disso, o Visual Studio Code se integra perfeitamente ao ecossistema do React, conforme mencionado por Jordan Walke, um dos desenvolvedores principais do React. Sua extensibilidade permite a incorporação de plug-ins e extensões especializadas para React, facilitando o desenvolvimento de componentes reutilizáveis e a construção dinâmica da interface do usuário. Dessa forma, a escolha do Visual Studio Code para o desenvolvimento com React no projeto Comtur está alinhada com a busca por uma ferramenta que não apenas atenda às exigências técnicas, mas também proporcione uma experiência de codificação ágil e eficaz para a equipe de desenvolvimento.

## 6.1.4 PostgreSQL

Figura 11 – Logo do PostgreSQL



Fonte: Medium(2022)

A figura acima, contém a logo do PostgreSQL, que desempenha um papel crucial na gestão eficiente do banco de dados e na garantia da integridade dos dados. O PostgreSQL, como ressaltado por Dave Page, diretor do projeto PostgreSQL, é uma poderosa base de dados relacional de código aberto, reconhecida por sua confiabilidade e recursos avançados.

A escolha do PGAdmin como ferramenta de administração do PostgreSQL é respaldada pela sua interface intuitiva e recursos abrangentes. Conforme afirmado por Joe Conway, membro sênior da comunidade PostgreSQL, "o PGAdmin é uma ferramenta essencial para administrar o PostgreSQL, permitindo visualizar e manipular dados, criar consultas complexas e gerenciar esquemas de banco de dados de maneira eficaz."

A escolha do PostgreSQL e do PGAdmin no ESTAGIOTECH é embasada na necessidade de um sistema robusto de gerenciamento de banco de dados e de uma ferramenta intuitiva para administrar dados de maneira eficiente. Essa combinação, respaldada por especialistas renomados, assegura a confiabilidade e a eficácia na gestão do banco de dados, contribuindo para a estabilidade e integridade do sistema como um todo.

Em síntese, a arquitetura de software no projeto ESTAGIOTECH emerge como a espinha dorsal que sustenta a concepção, desenvolvimento e evolução contínua da aplicação. Inspirada pelas palavras de Grady Booch, que a define como a estrutura fundamental que traduz os requisitos em uma representação tangível, e reforçada por especialistas como Martin Fowler e Ralph Johnson, a arquitetura no ESTAGIOTECH não é apenas uma questão de organização do código, mas sim uma estratégia indispensável para lidar com a complexidade inerente ao desenvolvimento de software.

A gestão eficiente do banco de dados, conduzida pelo PostgreSQL e administrada pelo PGAdmin, destaca a importância de escolhas estratégicas em todas as fases do desenvolvimento. Ao seguir os princípios de arquitetura delineados por especialistas renomados, o ESTAGIOTECH assegura uma base sólida para a expansão futura, a manutenção eficaz e a sustentabilidade do sistema. Nesse contexto, a arquitetura de software não é apenas uma camada técnica, mas sim um alicerce vital que guia o projeto em direção à eficiência, escalabilidade e sucesso duradouro.

## 6.2 Telas do sistema

*Colocar as telas desenvolvidas (figuras) e explicar sua função e funcionamento*

# CONCLUSÃO

O desenvolvimento e implementação do sistema de contratação e gestão de estagiários para os alunos da Fatec Jales representam um avanço nos recursos internos da instituição, otimizando tempo e diminuindo a burocracia. Durante a concepção deste projeto, evidenciou-se não apenas a dificuldade no recrutamento de novos talentos, mas também a escassez de mão de obra para concretizar esta iniciativa.

Os resultados obtidos nas fases de entrevista revelaram a necessidade concreta de continuidade desse sistema. Ele não só supre as demandas específicas da Fatec Jales, mas também estabelece um modelo replicável para outras instituições de ensino superior. A expectativa de aceitação positiva por parte dos alunos e docentes, aliada à redução dos problemas relacionados à documentação e alocação, reforça a relevância prática deste projeto.

Portanto, com base nos resultados apresentados, conclui-se que o sistema de gestão de contratos proposto não apenas atende às necessidades da Fatec Jales, mas também representa uma solução viável e eficaz para aprimorar a qualidade do processo de contratação, organização e alocação de novos estagiários. Recomenda-se a plena implementação do sistema, utilizando as lições aprendidas durante o desenvolvimento, e sugere-se que outras instituições de ensino considerem a adaptação desse modelo para otimizar seus próprios processos acadêmicos. Este trabalho contribui não apenas para o avanço tecnológico na gestão educacional, mas também para o aprimoramento contínuo da experiência do aluno no ambiente acadêmico.

# REFERÊNCIAS

ASFHAL, C. R. **Gestão de segurança do trabalho e de saúde ocupacional**. São Paulo: Reichmann & Autores, 2005.

BEAIRD, J. Princípios do Web Design Maravilhoso. Rio de Janeiro: Altabooks, 2008.

CHIAVENATO, I. **Gestão de pessoas**: o novo papel dos recursos humanos nas organizações. 3. ed. Rio de Janeiro: Elsevier, 2008.

COSTA, H. J. Acidentes do trabalho: teremos nova lei acidentária?. **Jus Navigandi**, Teresina, v. 9, n. 664, 1 maio 2005. Disponível em: <http://jus2.uol.com.br/doutrina/texto.asp?id=6662>. Acesso em: 1 jul. 2010.

DELIBERATO, P. C. P. **Fisioterapia preventiva**: fundamentos e aplicações. São Paulo: Manole, 2002.

DIAS, E. C.; MELO, E. M. de. Políticas públicas em saúde e segurança no trabalho. In: MENDES, R. (Org.). **Patologia do trabalho**. 2. ed. atual. e ampl. São Paulo: Atheneu, 2007. v. 2. p. 1683-1720.

GUEDES, Gilleanes T. A. **UML Uma Abordagem prática**, 3 ed. São Paulo: Novatec, 2008.

GUEDES, Gilleanes T. A. **UML 2 Uma Abordagem prática**, São Paulo: Novatec, 2009.

WATRALL, E; SIARTO, J. Use A Cabeça! **Web Design.** Alta Books, 2009.

UNGER, Russ; CHANDLER, Carolyn. **O guia para projetar UX**. 1. Ed. São Paulo: Alta Books, 2009.

NORMAN, D. A. (2002). **"The Design of Everyday Things."** Basic Books.

NIELSEN, J. (2006). **"Ten Usability Heuristics."**

MACHADO, F. N. R. **Banco de dados**: projeto e implementação. 4. Ed. São Paulo: Érica, 2020.

JOHNSON, R., GAMMA, E., HELM, R., & VLISSIDES, J. (1994). "**Design Patterns: Elements of Reusable Object-Oriented Software."** Addison-Wesley.

FOWLER, M. (2003). **"Patterns of Enterprise Application Architecture."** Addison-Wesley.

CWORK SISTEMAS (2021). **Logo C#.** < https://becode.com.br/wp-content/plugins/media-ace/assets/hotlink-placeholder.png >. Acesso em: 02 dez.2023

1000 Logos (2023). **Logo Visual Studio.** < https://1000logos.net/wp-content/uploads/2023/04/Visual-Studio-logo-500x281.png >. Acesso em: 02 dez.2023

Medium (2019). **Logo Visual Studio.** < https://miro.medium.com/v2/resize:fit:720/format:webp/1\*cn\_XBD307E3lObHk511Qqg.png >. Acesso em: 02 dez.2023

Medium (2022). **Logo PgAdmin.** < https://miro.medium.com/v2/resize:fit:720/format:webp/1\*f2Hiyitx0hHHAIDKc-SnoA.png>. Acesso em: 02 dez.2023