

Sistema de Controle de Ciclo de Vida do Funcionário com PDI

Funcionalidade: Gestão de Ciclo de Vida do Funcionário (GCVF) – Cadastro de Funcionários, Competências, PDI e Ciclo de Feedbacks.

Informações

Projeto/Requisição/Incidente	Sistema de Controle de Ciclo de Vida do Funcionário com PDI
Sistema	Gestão de Ciclo de Vida do Funcionário (GCVF)
Módulo	Cadastro de Funcionários, Gestão de Competências, PDI e Ciclo de Feedbacks
Objetivo	Criar um sistema escalável e flexível para gerenciar o ciclo de vida dos funcionários de uma empresa de TI. Os requisitos funcionais incluem o cadastro detalhado de dados de funcionários (incluindo níveis: Estagiário, Júnior, Pleno e Sênior, e área de atuação: Técnico ou Negócios) e stacks tecnológicas com níveis de conhecimento, a implementação completa de PDIs e o gerenciamento robusto de feedbacks (180° e 360° com diferentes tipos de dados), além de perfis de usuário com controle de acesso granular e funcionalidades adicionais como gestão de perguntas e sistema de pontuação. A solução visa otimizar a gestão de talentos, promover a evolução contínua dos colaboradores e a retenção de talentos, suportar o crescimento da empresa e integrar as necessidades de múltiplas entidades (FIERGS, Sesi, SENAI, IEL, CIERGS) em uma arquitetura web responsiva.
Analista Responsável	Guilherme Polano
Usuário Responsável	GINFO
Data do Aceite	A definir

Histórico

Versão	Data	Autor/Revisor	Descrição
1.0	28/01/2026	Guilherme Polano	Criação do Documento Preliminar
1.1	28/01/2026	Guilherme Polano	Revisão e adição de detalhes arquiteturais, de escalabilidade e objetivos de Retenção de Talentos e Evolução de Funcionários.
1.2	28/01/2026	Guilherme Polano	Reestruturação do documento para detalhamento completo dos requisitos funcionais e não funcionais, removendo "Arquitetura Hexagonal" do nome do projeto e movendo os detalhes para os requisitos não funcionais.

SUMÁRIO

Sumário

Sistema de Controle de Ciclo de Vida do Funcionário com PDI	1
Informações	1
Histórico.....	1
1 Visão Geral.....	3
2 Contexto Atual e Desafios	3
3 Requisitos Funcionais Detalhados	3
3.1 Cadastro de Funcionários.....	3
3.2 PDI e Feedbacks.....	4
3.3 Perfis de Usuário	5
3.4 Funcionalidades Adicionais	5
4 Requisitos Não Funcionais Detalhados	6
4.1 Escalabilidade.....	6
4.2 Segurança.....	6
4.3 Usabilidade	6
4.4 Manutenção e Evolução.....	7
4.5 Arquitetura e Tecnologias	7
4.6 Multi-entidade e Multi-setor.....	7
5 Estruturação de User Stories (Exemplos BDD)	7
5.1 Exemplo 1: Cadastro de Funcionário.....	7
5.2 Exemplo 2: Visualização de Stacks Tecnológicas.....	8
5.3 Exemplo 3: Acesso Restrito por Entidade/Setor	8
6 Anexos.....	9

Objetivo

Criar um sistema escalável e flexível para gerenciar o ciclo de vida dos funcionários de uma empresa de TI. Os requisitos funcionais incluem o cadastro detalhado de dados de funcionários (incluindo níveis: Estagiário, Júnior, Pleno e Sênior, e área de atuação: Técnico ou Negócios) e stacks tecnológicas com níveis de conhecimento, a implementação completa de PDIs e o gerenciamento robusto de feedbacks (180° e 360° com diferentes tipos de dados), além de perfis de usuário com controle de acesso granular e funcionalidades adicionais como gestão de perguntas e sistema de pontuação. A solução visa otimizar a gestão de talentos, promover a evolução contínua dos colaboradores e a retenção de talentos, suportar o crescimento da empresa e integrar as necessidades de múltiplas entidades (FIERGS, Sesi, SENAI, IEL, CIERGS) em uma arquitetura web responsiva.

1 Visão Geral

O presente documento detalha a proposta para o desenvolvimento de um sistema robusto e adaptável, focado no gerenciamento completo do ciclo de vida dos funcionários de uma empresa de TI com mais de 60 colaboradores. A iniciativa visa preencher uma lacuna na gestão de talentos, oferecendo ferramentas para um Plano de Desenvolvimento Individual (PDI) personalizado e um ciclo de feedbacks estruturado (180° e 360°). O sistema terá um papel fundamental na evolução profissional dos colaboradores e na retenção de talentos, criando um ambiente de crescimento e reconhecimento.

As funcionalidades principais do sistema incluirão o cadastro detalhado de funcionários (incluindo níveis: Estagiário, Júnior, Pleno e Sênior, e área de atuação: Técnico ou Negócios) com suas stacks tecnológicas e níveis de conhecimento, a configuração e acompanhamento de PDIs e diferentes modalidades de feedback (técnicos, comportamentais, 1 a 1), perfis de usuário distintos (Funcionário, Coordenador, Gerente, Administrador) com controle de acesso granular, e funcionalidades adicionais como a gestão de categorias e perguntas para avaliações e um sistema de pontuação para funcionários.

2 Contexto Atual e Desafios

Atualmente, a gestão do ciclo de vida dos funcionários, PDI e processos de feedback no setor de TI com mais de 30 colaboradores carecem de uma ferramenta integrada e padronizada. As informações podem estar dispersas em planilhas, documentos avulsos ou sistemas isolados, dificultando:

- O acompanhamento efetivo do desenvolvimento individual dos colaboradores.

- A padronização e a execução dos ciclos de feedback.

- A identificação precisa das stacks tecnológicas e níveis de proficiência da equipe.

- A análise de dados agregados sobre o desempenho e o desenvolvimento dos talentos.

- A gestão de diferentes entidades e setores de forma centralizada, mas com autonomia de visualização.

A falta de uma solução dedicada gera ineficiências operacionais, dificuldades na tomada de decisões estratégicas sobre o desenvolvimento de equipes e um potencial desperdício de talentos, impactando negativamente a evolução profissional e a retenção de funcionários.

3 Requisitos Funcionais Detalhados

Os requisitos funcionais detalham as capacidades e comportamentos específicos que o sistema deve apresentar para atender às necessidades dos usuários e do negócio.

3.1 Cadastro de Funcionários

Este módulo será a base do sistema, centralizando todas as informações cruciais sobre os colaboradores.

Dados Cadastrais Básicos:

- O sistema deve permitir o cadastro e a manutenção de dados básicos do funcionário, incluindo: nome completo, cargo, nível (Estagiário, Júnior, Pleno, Sênior), área de atuação (Técnico ou Negócios - por exemplo Analista Técnico, Analista de Negócios), departamento/setor, entidade (ex: FIERGS, Sesi, SENAI, IEL, CIERGS), data de admissão, data de nascimento, gênero, e-mail corporativo, telefone e status (ativo/inativo).

- Deve haver campos para identificação única do funcionário.

- A atualização desses dados deve ser controlada por perfis de usuário específicos (Coordenador, Gerente, Administrador).

Cadastro de Stacks Tecnológicas com Níveis de Conhecimento (0 a 5):

- O sistema deve permitir associar múltiplas stacks tecnológicas a cada funcionário.

- Cada stack tecnológica deve ter um nível de conhecimento atribuído, em uma escala de 0 a 5, onde:

 - Nível 0: Desconhece a tecnologia ou não possui experiência.

Nível 1: Conhecimento básico, familiaridade com conceitos e termos.

Nível 2: Conhecimento fundamental, capaz de realizar tarefas supervisionadas.

Nível 3: Proficiência, capaz de trabalhar de forma independente em projetos usando a tecnologia.

Nível 4: Especialista, capaz de resolver problemas complexos e orientar outros.

Nível 5: Referência, domínio completo, capaz de definir arquiteturas e influenciar a adoção da tecnologia na empresa.

Possibilidade de Cadastrar Múltiplos Níveis de Tecnologia:

O sistema deve suportar uma categorização hierárquica das tecnologias, permitindo o cadastro de:

Categorias de Tecnologia: Ex: Desenvolvimento Frontend, Backend, Banco de Dados, DevOps, QA.

Tecnologias/Linguagens: Ex: JavaScript, Python, Java, C#, SQL, Docker.

Frameworks/Ferramentas: Ex: React, Angular, Spring Boot, Nest.js, Django, Kubernetes, Selenium.

Cada nível (linguagem, framework, etc.) deve poder ter seu próprio nível de conhecimento associado ao funcionário.

Deve ser possível adicionar ou remover stacks e seus níveis a qualquer momento.

3.2 PDI e Feedbacks

Este módulo é o coração do sistema para o desenvolvimento contínuo dos colaboradores.

Configuração de Feedbacks 180° e 360° com Periodicidade Variável:

O sistema deve permitir a configuração de ciclos de feedback com periodicidade flexível (mensal, bimestral, trimestral, semestral, anual).

Deve ser possível configurar o tipo de feedback:

Feedback 180°: Envolve a autoavaliação do funcionário e a avaliação de seu gestor direto.

Feedback 360°: Inclui autoavaliação, avaliação do gestor, avaliação de colegas (pares) e, se aplicável, de subordinados.

A configuração do fluxo de participantes e da periodicidade deve ser gerenciável por Coordenadores, Gerentes e Administradores.

Feedbacks com Três Tipos de Dados:

Feedback Técnico:

Deve conter perguntas específicas sobre as stacks tecnológicas do funcionário e sua aplicação em projetos.

Deve avaliar a capacidade de resolução de problemas técnicos, qualidade do código, entre outros.

Feedback Comportamental:

Deve conter perguntas relacionadas a soft skills como comunicação, trabalho em equipe, proatividade, liderança, adaptabilidade, resolução de conflitos e cultura da empresa.

Conversa 1 a 1:

Deve haver um espaço dedicado para o registro de anotações, pautas e resultados de conversas individuais entre o funcionário e seu gestor.

Permite documentar objetivos de PDI, desafios, planos de ação e suporte necessário.

Autoavaliação com Perguntas Pré-cadastradas e Possibilidade de Novas Perguntas:

O sistema deve disponibilizar um banco de perguntas categorizadas para autoavaliação.

Coordenadores e Gerentes devem ter a funcionalidade de criar, editar e excluir perguntas, bem como suas categorias, para personalizar os formulários de feedback.

As perguntas podem ser de múltipla escolha, escala numérica (ex: 1 a 5), ou texto livre.

Relatórios de Feedback com Acordos, Pontos Positivos e a Melhorar:

Após a conclusão de um ciclo de feedback, o sistema deve gerar relatórios consolidados para o funcionário e seu gestor.

Esses relatórios devem apresentar claramente:

Pontos Positivos: Destaque para as forças e contribuições do funcionário.

Pontos a Melhorar: Áreas identificadas para desenvolvimento.

Acordos e Planos de Ação: Objetivos de desenvolvimento específicos, ações a serem tomadas (PDI) e prazos definidos.

Os relatórios devem ser visualmente claros e acessíveis apenas às partes envolvidas.

Acompanhamento de PDI:

O sistema deve permitir o registro e o acompanhamento de Planos de Desenvolvimento Individual (PDI) vinculados aos resultados dos feedbacks.

Cada PDI deve conter: objetivos SMART (Specific, Measurable, Achievable, Relevant, Time-bound), ações propostas, recursos necessários, prazos e status de progresso.

Funcionários e gestores devem poder visualizar e atualizar o status do PDI.

3.3 Perfis de Usuário

A segregação de responsabilidades é essencial para a segurança, organização do fluxo de trabalho e controle de acesso.

Funcionário:

Autoavaliação: Capacidade de realizar suas autoavaliações dentro dos ciclos de feedback configurados.

Visualização de Feedbacks e PDI: Acesso exclusivo aos seus relatórios de feedback (180°, 360°) e ao seu Plano de Desenvolvimento Individual, incluindo status de progresso e acordos.

Visualização de Dados Próprios: Acesso às suas informações de cadastro e às suas stacks tecnológicas com níveis de conhecimento.

Não pode: Visualizar informações de outros funcionários, editar dados de outros, configurar ciclos de feedback.

Coordenador:

Gestão de Funcionários: Cadastro e edição de dados cadastrais e stacks tecnológicas dos funcionários de sua equipe e setor.

Gestão de Feedbacks: Configuração e acompanhamento dos ciclos de feedback para os funcionários de sua equipe. Pode iniciar, monitorar e finalizar ciclos.

Avaliação de Liderados: Realiza avaliações dos funcionários de sua equipe (conforme configuração de feedback).

Gestão de Perguntas: Criação, edição e exclusão de categorias e perguntas para autoavaliação e feedbacks.

Visibilidade: Acesso apenas aos dados de funcionários que pertencem à sua entidade e setor.

Gerente:

Funcionalidades de Coordenador: Possui todas as funcionalidades do perfil de Coordenador.

Visão Ampliada: Acesso a dados e relatórios agregados de todas as equipes sob sua gerência direta ou indireta dentro de sua entidade.

Aprovações: Pode ter permissões adicionais para aprovar configurações de feedback ou PDIs, dependendo da governança.

Visibilidade: Acesso apenas aos dados de funcionários que pertencem à sua entidade e seus setores subordinados.

Administrador (Global):

Controle Total: Acesso irrestrito a todas as funcionalidades do sistema.

Gerenciamento de Usuários e Perfis: Criação, edição, exclusão de usuários e atribuição de perfis de acesso.

Configurações Globais: Definição de parâmetros globais do sistema (ex: tipos de feedback, escalas de proficiência, categorias de tecnologia).

Acesso a Todos os Dados: Visão completa e consolidada de todos os funcionários, PDIs, feedbacks e dados de todas as entidades (FIERGS, Sesi, SENAI, IEL, CIERGS) e seus setores.

Auditoria: Acesso completo a logs de auditoria do sistema.

3.4 Funcionalidades Adicionais

Estas funcionalidades enriquecem o sistema e o tornam mais estratégico.

Cadastro de Categorias e Perguntas para Autoavaliação e Feedbacks:

Permite a criação e organização de um banco de dados de perguntas, classificadas por categoria (técnica, comportamental, etc.), que podem ser reutilizadas em diferentes formulários de feedback e autoavaliação.

Coordenadores, Gerentes e Administradores podem gerenciar esse banco de perguntas.

Sistema de Pontuação para Funcionários Baseado nos Feedbacks:

Um mecanismo configurável para atribuir e calcular pontuações para funcionários com base em critérios definidos nos feedbacks (ex: média das avaliações, cumprimento de metas do PDI, evolução de nível em stacks).

A pontuação pode ser exibida de forma privada (para funcionário e gestor) ou agregada (para gerentes e administradores) para análise de desempenho e reconhecimento.

Matriz de Habilidades (Skills Matrix):

Funcionalidade para visualizar graficamente ou em tabela as stacks tecnológicas e os níveis de proficiência da equipe ou de toda a organização.

Permite identificar gaps de conhecimento, talentos específicos e planejar treinamentos ou alocação de projetos.

Filtros por entidade, setor, cargo, nível, área de atuação e tecnologia.

4 Requisitos Não Funcionais Detalhados

Os requisitos não funcionais descrevem como o sistema deve funcionar, focando em atributos de qualidade, restrições e características arquitetônicas.

4.1 Escalabilidade

O sistema deve ser projetado para suportar o crescimento contínuo da empresa e suas entidades, acomodando um número crescente de usuários (funcionários, coordenadores, gerentes, administradores) e um volume maior de dados (registros de feedback, PDIs, históricos de competências) sem degradação perceptível de desempenho.

A arquitetura deve permitir a expansão horizontal de serviços e a otimização de consultas ao banco de dados para garantir respostas rápidas mesmo sob alta carga.

4.2 Segurança

Autenticação:

O sistema deve exigir autenticação segura para todos os usuários.

Será prioritária a integração com sistemas de autenticação corporativos existentes (ex: Active Directory/LDAP) para simplificar o gerenciamento de credenciais.

Deve suportar autenticação multifator (MFA) para maior segurança.

Autorização e Controle de Acesso Baseado em Papéis (RBAC):

O acesso às funcionalidades e aos dados deve ser estritamente controlado com base nos perfis de usuário definidos (Funcionário, Coordenador, Gerente, Administrador).

A granularidade do acesso deve considerar a hierarquia de entidades e setores (multi-entidade/multi-setor), garantindo que usuários vejam apenas as informações pertinentes à sua área de atuação, exceto o administrador global.

Criptografia de Dados:

Todos os dados sensíveis (informações de funcionários, feedbacks) devem ser criptografados tanto em trânsito (utilizando HTTPS/TLS) quanto em repouso (no banco de dados).

Auditoria e Logs:

O sistema deve registrar logs detalhados de todas as operações críticas (cadastro, edição, exclusão de dados, acessos, tentativas de acesso falhas) para fins de auditoria e rastreabilidade.

Os logs devem ser protegidos contra adulteração e acessíveis apenas por Administradores.

Conformidade:

O sistema deve estar em conformidade com as leis e regulamentações de proteção de dados, como a LGPD (Lei Geral de Proteção de Dados) no Brasil.

4.3 Usabilidade

Interface Intuitiva: A interface do usuário (UI) deve ser limpa, moderna e intuitiva, minimizando a curva de aprendizado e facilitando a navegação para todos os perfis de usuário.

Experiência do Usuário (UX): O design deve focar na experiência do usuário, oferecendo fluxos lógicos e eficientes para a execução das tarefas, como preenchimento de feedbacks e acompanhamento de PDIs.

Responsividade: O sistema deve ser totalmente responsivo, garantindo uma experiência de usuário otimizada e consistente em uma ampla gama de dispositivos e tamanhos de tela (desktops, laptops, tablets, smartphones).

Acessibilidade: Deve-se buscar conformidade com padrões de acessibilidade web (ex: WCAG) para garantir que o sistema possa ser utilizado por pessoas com diferentes capacidades.

4.4 Manutenção e Evolução

Manutenibilidade: O código-fonte deve ser limpo, bem documentado e seguir padrões de codificação estabelecidos, facilitando a identificação e correção de bugs e a implementação de melhorias.

Extensibilidade: A arquitetura e o design do sistema devem permitir a fácil adição de novas funcionalidades e módulos sem a necessidade de refações extensas na lógica existente.

Robustez: O sistema deve ser resiliente a falhas, com tratamento de erros adequado e mecanismos de recuperação.

4.5 Arquitetura e Tecnologias

Arquitetura Hexagonal (Ports and Adapters):

O sistema será desenvolvido seguindo os princípios da Arquitetura Hexagonal, também conhecida como Ports and Adapters. Esta abordagem garante uma forte separação entre a lógica de negócios central (o "domínio") e as preocupações de infraestrutura (UI, bancos de dados, serviços externos).

O domínio se comunica com o mundo exterior através de "portas" (interfaces), e "adaptadores" implementam essas portas para tecnologias específicas. Isso promove:

Independência: O domínio é agnóstico em relação à tecnologia, tornando-o imune a mudanças externas.

Testabilidade: A lógica de negócios pode ser testada isoladamente, com mock-ups simples dos adaptadores.

Flexibilidade: Permite substituir ou adicionar facilmente componentes de infraestrutura (ex: trocar de banco de dados ou framework UI) sem afetar o core do sistema.

Tecnologias Web:

Front-end: Será utilizado Next.js, um framework React, para desenvolver a interface do usuário. Next.js oferece Server-Side Rendering (SSR) e Static Site Generation (SSG), o que contribui para a performance, SEO e uma experiência de usuário fluida em aplicações web responsivas.

Back-end: Será utilizado Nest.js, um framework Node.js baseado em TypeScript. Inspirado no Angular, Nest.js fornece uma arquitetura robusta e modular para construir APIs eficientes e escaláveis, ideal para gerenciar a lógica de negócios e a interação com o banco de dados.

Banco de Dados Relacional:

Ambiente On-premises: Para implantações em infraestrutura local, o sistema utilizará o Oracle Database. Esta escolha se baseia na sua robustez, funcionalidades avançadas de segurança, alta disponibilidade e performance para cargas de trabalho empresariais.

Ambiente em Nuvem: Para implantações em ambientes de nuvem, a escolha será o MariaDB. Por ser um banco de dados relacional de código aberto, o MariaDB oferece flexibilidade, custo-benefício e é amplamente suportado por provedores de nuvem, garantindo escalabilidade e facilidade de gerenciamento.

4.6 Multi-entidade e Multi-setor

O sistema deve ser capaz de gerenciar dados e usuários para múltiplas entidades (FIERGS, Sesi, SENAI, IEL, CIERGS) e seus respectivos setores.

Segregação de Dados: A arquitetura de dados deve garantir a segregação lógica ou física das informações, de modo que cada entidade e setor opere com seus próprios conjuntos de dados, visíveis apenas aos usuários autorizados.

Controle de Acesso: O mecanismo de autorização (RBAC) deve ser configurado para impor que:

Usuários com perfil de "Funcionário", "Coordenador" ou "Gerente" só possam visualizar e interagir com dados (funcionários, PDIs, feedbacks) que pertençam à sua própria entidade e, no caso de Coordenadores/Gerentes, ao seu setor ou equipes subordinadas.

Apenas o perfil de "Administrador (Global)" terá permissão para visualizar e gerenciar dados de todas as entidades e setores, permitindo uma visão estratégica e governança centralizada.

5 Estruturação de User Stories (Exemplos BDD)

Os requisitos funcionais serão detalhados em User Stories, seguindo o padrão Behavior-Driven Development (BDD), com Critérios de Aceite claros para guiar o desenvolvimento e os testes.

5.1 Exemplo 1: Cadastro de Funcionário

Como um Coordenador, Eu quero cadastrar um novo funcionário, Para que suas informações básicas sejam registradas no sistema.

Critérios de Aceite:

Cenário 1: Cadastro bem-sucedido com dados válidos.

Dado que estou logado como Coordenador da FIERGS - GINFO,

E navego para a tela de "Cadastro de Funcionário",

Quando preencho todos os campos obrigatórios (Nome, Cargo, Nível, Área de atuação, Entidade: FIERGS, Setor: GINFO, Data de Admissão, E-mail) com dados válidos,

E clico em "Salvar",

Então o funcionário é registrado com sucesso,

E sou redirecionado para a tela de "Detalhes do Funcionário" do recém-cadastrado,

E uma notificação de sucesso é exibida.

Cenário 2: Tentativa de cadastro com campos obrigatórios vazios.

Dado que estou logado como Coordenador da FIERGS - GINFO,

E navego para a tela de "Cadastro de Funcionário",

Quando deixo um ou mais campos obrigatórios vazios,

E clico em "Salvar",

Então o sistema exibe mensagens de erro indicando quais campos devem ser preenchidos,

E o funcionário não é registrado.

5.2 Exemplo 2: Visualização de Stacks Tecnológicas

Como um Funcionário, Eu quero visualizar minhas stacks tecnológicas cadastradas com meu nível de conhecimento, Para que eu possa acompanhar minhas competências e planejar minha evolução.

Critérios de Aceite:

Cenário 1: Visualização das stacks tecnológicas próprias.

Dado que estou logado como Funcionário (João da Silva),

E minhas stacks tecnológicas (ex: JavaScript Nível 3, React Nível 2) estão cadastradas,

Quando navego para a seção "Minhas Competências",

Então consigo ver todas as minhas tecnologias, linguagens e frameworks cadastrados,

E para cada um, consigo ver meu nível de conhecimento atribuído (0 a 5).

5.3 Exemplo 3: Acesso Restrito por Entidade/Setor

Como um Coordenador, Eu quero visualizar apenas os funcionários da minha entidade e setor, Para garantir a privacidade e a organização dos dados.

Critérios de Aceite:

Cenário 1: Coordenador visualiza apenas sua equipe.

Dado que estou logado como Coordenador da FIERGS - GINFO,

E existem funcionários cadastrados na FIERGS - GINFO e na Sesi - RH,

Quando acesso a lista de "Funcionários",

Então só consigo ver os funcionários que pertencem à FIERGS - GINFO,

E não consigo ver os funcionários da Sesi - RH.

Cenário 2: Administrador global visualiza todas as equipes.

Dado que estou logado como Administrador global,

E existem funcionários cadastrados em todas as entidades e setores,

Quando acesso a lista de "Funcionários",

Então consigo ver todos os funcionários de todas as entidades e setores.

6 Anexos

Diagramas de Arquitetura (a serem desenvolvidos)

Modelagem de Dados (a ser desenvolvida)

Planilha de Priorização de Funcionalidades (a ser desenvolvida)

Resultados da Pesquisa de Mercado (a serem compilados)