Documento de Requisitos de Software

Projeto: Sistema de Gestão dos Cursos de Capacitação SUKATECH – Programando o Futuro

Guilherme Silva Virgilli

Joyce Beatriz Ferreira da Costa Silva

Mariana Gonçalves Landi

Ubiratan Alves Paniago Filho

Delvo Resende

Versão 1.0  
Data: Setembro/2025

Sumário

[**1. Introdução**](#_6b4mf57i6s0z) **2**

[**2. Visão Geral do Produto**](#_alf9pneoq0wq) **2**

[**3. Requisitos Funcionais**](#_skwci01a79ru) **3**

[**4. Requisitos Não Funcionais**](#_swdm52my13iv) **8**

[**5. Restrições de Design e Tecnológicas**](#_sav685u61mu2) **10**

[**6. Modelagem**](#_8232z5cozztt) **10**

[**7. Critérios de Aceite e Testes**](#_4e9bic29fa1x) **18**

[**8. Riscos e Impactos**](#_34c4k055ol0x) **19**

[**9. Anexos**](#_bdqd54x9l0xo) **19**

## Introdução

* 1. Propósito (descrever o objetivo do sistema e deste documento)
  2. Escopo: descrição simplificada do que o sistema vai fazer

## Visão Geral do Produto

* 1. Usuários e Personas
     1. Administrador: controla sistemas e relatórios
     2. Alunos: gestão de cadastro, matrículas, desistências
     3. Instrutores: gestão do cadastro, gestão pedagógica (frequência, rendimento)
     4. Turmas: operacionalização dos cursos (vagas, calendário, atribuição de instrutores para ministrar trilha/curso, alunos)
     5. Cursos/Trilhas: organização de conteúdo e os objetivos de aprendizado
     6. Inscrição e Matrículas: link entre o aluno e a turma
     7. Home: centralização dos acessos (simplificar ao máximo - poucas funcionalidades) orientação por perfis

## Requisitos Funcionais

| **Requisitos Funcionais - Módulo ALUNO** | | | |
| --- | --- | --- | --- |
| **Código** | **Requisito Funcional (User Story)** | **Critério de Análise** | **Prioridade** |
| RF01-A | Como **administrador,** quero cadastrar um aluno informando dados pessoais (nome, e-mail, telefone, endereço, data nascimento), para registrar sua participação na turma do curso | - O sistema deve validar o CPF único.  - E-mail deve ser válido  - Campos obrigatórios devem ser preenchidos  - Após salvar o aluno deve receber o ID único (matrícula) | Alta |
| RF02-A | Como **administrador** quero consultar os dados de um aluno por nome, CPF ou ID (matrícula), para acessar rapidamente as informações do aluno | - O sistema deve permitir busca por múltiplos critérios.  - Listar resultados em formato de tabela.  - Permitir filtros e ordenação | Alta |
| RF03-A | Como **administrador** quero editar os dados de um aluno, para corrigir ou atualizar informações pessoais | - O sistema deve exibir dados atuais  - Validar novamente os campos obrigatórios  - Histórico de alterações deve ser armazenado | Alta |
| RF04-A | Como **administrador,** quero excluir o cadastro de um aluno, para remover registros incorretos ou duplicados | - Exclusão deve solicitar confirmação.  - Não deve ser possível excluir o aluno com matrícula ativa no curso.  - Registro pode ser marcado como “inativo” em vez de removido do sistema | Média |
| RF05-A | Como **administrador,** quero listar todos os alunos cadastrados, para gerar relatórios e visualizar estatísticas | - Deve listar tabela paginada  - permitir exportar o arquivo em pdf  - Deve mostrar total de alunos ativos e inativos | Média |
| RF06-A | Como **administrador,** quero vincular o aluno a uma turma já cadastrada | -O sistema deve verificar se a turma está ativa e com vagas disponíveis  -Deve impedir matrícula duplicada na mesma turma  -Após a matrícula, aluno aparece na lista da turma | Alta |
| RF07-A | Como **administrador,** quero desvincular aluno da turma permitindo o cancelamento da matrícula de um aluno em determinada turma | -Deve registrar o motivo e a data de desvinculação  -Deve atualizar o número de vagas disponíveis | Alta |
| RF08-A | Como **administrador,** quero consultar turmas do aluno permitindo visualizar todas as turmas que o aluno está matriculado | -Deve exibir o nome da turma, curso, status da matrícula e o progresso do aluno.  -Deve permitir exportar no formato de relatório (pdf/excel) | Alta |
| RF09-A | Como **administrador,** quero emitir relatórios de alunos ativos, turma vinculadas e desempenho | -Deve exibir lista por turma, curso ou trilha.  -Deve permitir exportação do relatório em pdf/excel | Média |

| **Requisitos Funcionais - Módulo INSTRUTOR** | | | |
| --- | --- | --- | --- |
| **Código** | **Requisito Funcional (User Story)** | **Critério de Análise** | **Prioridade** |
| RF01-I | Como **administrador,** quero cadastrar novo instrutor informando os dados pessoais: nome do instrutor, cpf, e-mail, endereço, telefone, data de nascimento para registrar sua disponibilidade em cursos e turmas | **-**Sistema deve validar CPF único  -E-mail deve ser válido  -Campos obrigatórios (inclusive formação) devem ser preenchidos  - Após salvar, instrutor deve receber um ID único | Alta |
| RF02-I | Como **administrador,** quero consultar os dados de um instrutor por nome, CPF, para acessar rapidamente suas informações | **- O** sistema deve permitir busca por múltiplos critérios.  -Listar resultados em formato de tabela  - Permitir filtros e ordenação | Alta |
| RF03-I | Como **administrador,** quero editar os dados de um instrutor, para corrigir ou atualizar informações pessoais e profissionais do instrutor | **-** O sistema deve exibir dados atuais.  - Validar novamente os campos obrigatórios.  - Histórico de alterações deve ser armazenado | Alta |
| RF04-I | Como **administrador,** quero excluir o cadastro de um instrutor, para remover registros incorretos ou duplicados | **-** Exclusão deve solicitar confirmação.  - Não deve ser possível excluir instrutor vinculado a turmas ativas  - Registro pode ser marcado como ‘inativo’ e não ser apagado do banco de dados | Média |
| RF05-I | Como **administrador,** quero listar todos os instrutores cadastrados, para gerar relatórios e visualizar estatísticas | **-** Deve listar em tabela paginada.  - Permitir exportação em pdf | Média |

| **Requisitos Funcionais - Módulo TURMA** | | | |
| --- | --- | --- | --- |
| **Código** | **Requisito Funcional (User Story)** | **Critério de Análise** | **Prioridade** |
| RF01-T | Como **administrador,** quero cadastrar uma turma para que ela fique disponível para inscrição de alunos. Informando os seguintes dados: nome da turma, turno, horário da turma, data de início da turma. | -Sistema deve permitir informar o nome do curso, período, quantidade de vagas e instrutor  - Após salvar, turma deve constar na listagem de turmas | Alta |
| RF02-T | Como **administrador,** quero visualizar informações completas de uma turma | -Deve exibir curso, instrutor, vagas, calendário, lista de alunos.  -Deve permitir consulta rápida por código ou nome de turma | Alta |
| RF03-T | Como **administrador,** quero alterar dados de uma turma cadastrada anteriormente | -Deve permitir edição de capacidade, instrutor, informações gerais, datas | Alta |
| RF04-T | Como **administrador,** quero excluir uma turma quando não for mais necessário armazená-la | -Verificar antes se há alunos matriculados, não existindo, poderá ser removida  -Emitir alerta de exclusão antes da execução da ação | Alta |
| RF05-T | Como **administrador,** quero listar todas as turmas cadastradas | -Listagem deve permitir filtros por curso, instrutor e status (ativa/concluída)  -Deve exibir quantidade de vagas e alunos matriculados | Alta |
| RF06-T | Como **administrador,** quero atribuir um instrutor a uma turma | - O sistema deve permitir selecionar instrutor ativo.  - Apenas instrutores cadastrados podem ser atribuídos.  - Deve garantir que o instrutor não esteja vinculado a duas turmas no mesmo horário. | Alta |
| RF07-T | Como **administrador,** quero gerenciar alunos da turma (adicionar/remover) | -Deve verificar limite de vagas  -Não deve permitir inscrições duplicadas | Alta |
| RF08-T | Como **administrador,** quero configurar número máximo de alunos por turma | -Deve permitir estabelecer o limite máximo da turma  -Deve recusar novas matrículas quando atingir a capacidade máxima da turma   |  | | --- | | Alta |
| RF09-T | Como **administrador,** quero visualizar agenda da turma | - Deve exibir dias, horários e local das aulas.  - Deve permitir impressão/exportação do calendário. | Média |
| RF10-T | Como **administrador,** quero emitir relatórios de presença e desempenho | - Relatório deve mostrar lista de alunos, frequência e médias.  - Deve permitir exportar em PDF/Excel. | Média |

| **Requisitos Funcionais - Módulo CURSO** | | | |
| --- | --- | --- | --- |
| **Código** | **Requisito Funcional (User Story)** | **Critério de Análise** | **Prioridade** |
| RF01-C | Como **administrador,** quero cadastrar cursos no sistema informando os seguintes dados: nome do curso, carga horária, horário programado, data de início. | - Deve conter título, descrição, carga horária, status (ativo/inativo).  - Após salvar, o curso deve aparecer na listagem. | Alta |
| RF02-C | Como **administrador,** quero visualizar todas as informações de um curso específico | - Deve exibir todos os dados cadastrados do curso.  - Deve permitir busca por título. | Alta |
| RF03-C | Como **administrador,** quero editar informações de um curso já cadastrado. | - Alterações devem ser registradas e refletidas imediatamente. | Alta |
| RF04-C | Como **administrador, quero r**emover um curso do sistema, desde que não esteja vinculado a turmas ativas. | - Bloquear exclusão caso existam turmas em andamento.  - Deve pedir confirmação antes da exclusão. | Alta |
| RF05-C | Como **administrador,** quero listar todos os cursos cadastrados (ativos e inativos) | - Deve permitir filtros por status ou outras opções pertinentes definidas no futuro  - Deve ter opção de exportação em PDF/Excel. | Alta |
| RF06-C | Como **administrador,** quero permitir que cada turma seja associada a um curso específico. | - Apenas cursos ativos podem ser vinculados.  - Cada turma deve ter referência clara ao curso correspondente. | Alta |
| RF07-C | Como **administrador,** quero listar todas as turmas já abertas para um curso. | - Listagem deve mostrar status da turma, instrutor responsável, período letivo e número de alunos matriculados | Alta |
| RF08-C | Como **administrador,** quero emitir relatórios de cursos ativos, turmas vinculadas e número de alunos. | - Relatórios por curso e status.  - Exportação em PDF/Excel. | Média |

## Requisitos Não Funcionais

Os atributos de qualidade de um software, neste caso, Sistema de Gestão de Cursos Sukatech - Programando o Futuro, visam garantir o sucesso do funcionamento do sistema. Eles definirão o que as funcionalidades devem realizar para atender as necessidades dos usuários.

* 1. **Desempenho**: é a rapidez com que os usuários podem interagir com a plataforma. Imagine um usuário tentando cadastrar o aluno para efetivar sua matrícula na turma de algum curso. A resposta aceitável tem a “métrica <2s em 95% das requisições”, e isto significa que, a grande maioria dessas ações deve ser praticamente instantâneas. Se o sistema reagir lentamente para cadastrar um único aluno vai gerar insatisfação e a experiência do usuário será qualificada ‘ruim’.
  2. **Segurança**: A segurança é um atributo crítico neste sistema. Há dados sensíveis armazenados em banco de dados, informações pessoais de alunos, instrutores (nomes, cpf, e-mails, etc). Sugere-se que use autenticação multifator (MFA) que adiciona uma camada extra de proteção garantindo que o acesso não será liberado ainda que a senha e usuário estejam comprometidos. O compliance com a LGPD é essencial e vai garantir que o tratamento desses dados esteja em conformidade com a lei, usuários estarão protegidos e os dados mantidos privados.
  3. **Usabilidade**: Aqui tratamos da facilidade de uso do sistema. O sistema deve ser intuitivo para todos os usuários independente do conhecimento e domínio de sistemas tecnológicos. O sistema deve permitir inclusive que usuários com alguma limitação visual, auditiva ou motora possam utilizar a plataforma sem barreiras, tornando seu uso inclusivo.
  4. **Confiabilidade**: Sistemas computadorizados confiáveis devem oferecer uma disponibilidade mínima de 99,9% que é um requisito alto, raramente deverá sair do ar. Para o administrador, usuário a confiabilidade é essencial, pois a qualquer momento será necessário cadastrar um aluno, instrutor, listar turmas, alunos, instrutores. Falhas frequentes prejudicam o processo de ensino e aprendizagem.
  5. **Escalabilidade e manutenibilidade**: Escalabilidade trata da capacidade de crescimento do sistema junto com a instituição. Se em algum momento do tempo houver necessidade de duplicar cursos, turmas, alunos, o sistema deve ser capaz de suportar esse aumento. Manutenibilidade trata da facilidade de fazer melhorias e correções eventuais no sistema. Um software bem estruturado permite que a equipe de desenvolvimento adicione novos recursos sem muito esforço de forma rápida e precisa, sem riscos eventuais, isso vai garantir que o software evolua ao longo do tempo.

## Restrições de Design e Tecnológicas

Linguagens, frameworks, banco de dados, padrões arquiteturais.

APIs e integrações obrigatórias.

Requisitos de infraestrutura.

## Modelagem

Diagrama ER - Entidade-Relacionamento Sukatech. Elaborado na ferramenta UML - EERCASE. É a identificação básica das entidades que compõem o Eixo 2 - Sukatech - Programando o Futuro - Capacitação.

É a representação das entidades do sistema e os relacionamentos entre elas, traduzindo as regras de negócios em um modelo visual

Curso: Representa um curso ou treinamento oferecido pelo programa.

Turma: Representa uma instância específica de um curso com alunos.

Aluno: Representa o estudante matriculado em uma turma.

Instrutor: Representa uma pessoa que ministra um curso.

CursoInstrutor: Entidade associativa para representar a associação N:N entre Curso e Instrutor

InstrutorTurma: Entidade associativa para representar a associação N:N entre Turma e Instrutor

Associação: A conexão entre as classes do diagrama UML são associações e os valores de cardinalidade (1, N, 1:N, N:N)

O projeto conceitual Sukatech de banco de dados foi criado com a plataforma EERCASE. Com ela é possível validar modelos usando a notação Modelo Entidade-Relacionamento (EER). Essa ferramenta foi criada no Centro de Informática (CIn) da Universidade Federal de Pernambuco (UFPE) e oferece suporte para geração de scripts de criação de tabelas e documentação dessas estruturas de dados. É uma ferramenta CASE (Computer-Aided Software Engineering - Engenharia de Software Auxiliada por Computador)

| Modelo Conceitual - Diagrama Entidade-Relacionamento (EER) |
| --- |
| Figura 1. Modelo Conceitual - Diagrama Entidade-Relacionamento (EER) |

Diagrama de Classes UML

É a representação das classes do sistema e as associações entre elas, traduzindo as regras de negócios em um modelo visual. Estão definidos no diagrama os atributos e as operações básicas para manipulação no banco de dados relacional. Traduz também as regras de negócios para persistência das informações. É o plano detalhado para a criação das tabelas: cada classe uma tabela, os atributos são as colunas e os relacionamentos definem as chaves estrangeiras que ligam com as chaves primárias das respectivas tabelas.

Curso: Representa um curso ou treinamento oferecido pelo programa.

Turma: Representa uma instância específica de um curso com alunos.

Aluno: Representa o estudante matriculado em uma turma.

Instrutor: Representa uma pessoa que ministra um curso.

CursoInstrutor: Entidade associativa para representar a associação N:N entre Curso e Instrutor

InstrutorTurma: Entidade associativa para representar a associação N:N entre Turma e Instrutor

O diagrama de classes UML foi construído com a ferramenta StarUML que possibilita criar, visualizar e manter diversos tipos de programas para modelagem de sistemas, como diagramas UML, SysUML e de Sequência. São várias funcionalidades que suportam a modelagem ágil, planejamento de projetos, visualização das funcionalidades de um sistema para usuários e equipes de desenvolvimento.

|  |
| --- |
| Figura 2. Modelo Lógico - Diagrama de Classes UML do Sistema de Gestão de Cursos Sukatech |

Diagramas UML (caso de uso).

Visão Geral do Sistema

O sistema foi projetado para gerenciar informações acadêmicas, abrangendo o controle de alunos, turmas, cursos e instrutores. O diagrama de casos de uso ilustra as funcionalidades disponíveis para os usuários do sistema, centralizadas em um único ator principal.

Atores

No diagrama, identificamos um ator principal:

Administrador: É o usuário principal do sistema. Ele possui permissão para executar todas as funcionalidades de cadastro, consulta, edição, associação e exclusão de dados nos diferentes módulos do sistema.

Modulos e Casos de Uso

O sistema é dividido em quatro módulos principais, cada um com um conjunto específico de funcionalidades (casos de uso) que o Administrador pode realizar.

1. Módulo Aluno

Este módulo concentra as funcionalidades relacionadas ao gerenciamento de dados dos alunos.

* Cadastrar aluno: Permite ao administrador registrar um novo aluno no sistema.
* Consultar dados: Possibilita a busca e visualização das informações de um aluno específico.
* Listar alunos: Exibe uma lista com todos os alunos cadastrados no sistema.
* Vincular aluno a turma: Permite associar um aluno já cadastrado a uma turma existente.
* Desvincular aluno da turma: Permite remover a associação de um aluno de uma turma.
* Consultar turmas do aluno: Permite visualizar todas as turmas em que um determinado aluno está matriculado.
* Excluir dados: Permite remover o registro de um aluno do sistema.
* Editar dados: Permite alterar as informações cadastrais de um aluno.
* Emitir relatório de alunos: Funcionalidade para gerar relatórios de alunos.
  + *Extensão (Extend):* Esta funcionalidade pode ser estendida para filtrar por status de aluno, incluindo "Ativos" e "Não ativos".

1. Módulo Turma

Este módulo é responsável pelo gerenciamento completo das turmas.

* Cadastrar turma: Permite ao administrador criar uma nova turma no sistema.
* Visualizar informações da turma: Permite consultar os detalhes de uma turma específica (instrutor, alunos, etc.).
* Alterar dados da turma: Permite editar as informações de uma turma já criada.
* Excluir turma: Permite remover o registro de uma turma do sistema.
* Listar dados: Exibe uma lista de todas as turmas cadastradas.
* Atribuir instrutor: Permite associar um instrutor a uma turma específica.
* Gerenciar alunos: Funcionalidade para administrar os alunos de uma turma.
  + *Inclusão (Include):* Esta funcionalidade inclui as ações de "Adicionar" e "Remover" alunos da turma.
* Configurar nº máx de aluno: Permite definir o número máximo de alunos para uma turma.
* Visualizar agenda: Permite consultar a agenda de aulas ou eventos da turma.
* Emitir relatórios: Permite gerar relatórios relacionados às turmas.

1. Módulo Curso

Este módulo gerencia as informações sobre os cursos oferecidos.

* Visualizar informações de um curso: Permite consultar os detalhes de um curso específico.
* Remover curso: Permite excluir o registro de um curso do sistema.
* Cadastrar cursos: Permite ao administrador registrar um novo curso.
* Editar info curso: Permite alterar as informações de um curso existente.
* Listar cursos: Exibe uma lista com todos os cursos disponíveis.
* Associar turma a um curso: Permite vincular uma ou mais turmas a um determinado curso.
* Listar todas as turmas já abertas para um curso: Permite visualizar todas as turmas que foram criadas para um curso específico.
* Emitir relatórios: Permite gerar relatórios relacionados aos cursos.

1. Módulo Instrutor

Este módulo é focado no gerenciamento dos dados dos instrutores.

* Cadastrar instrutor: Permite ao administrador registrar um novo instrutor no sistema.
* Consultar dados: Permite buscar e visualizar as informações de um instrutor específico.
* Editar dados: Permite alterar as informações cadastrais de um instrutor.
* Listar dados: Exibe uma lista com todos os instrutores cadastrados.
* Excluir dados: Permite remover o registro de um instrutor do sistema.

|  |
| --- |
| Figura 3 - Casos de Uso UML |

**Wireframes / protótipos de tela.**

[Protótipos Sukatech - FIGMA](https://www.figma.com/proto/afSQkcBVKj2KsrmcuzGMlc/Sukatech?node-id=14-45)

O Figma foi a ferramenta utilizada para a criação dos wireframes e protótipos de tela do sistema SUKATECH. Por ser uma plataforma online, ele permite a colaboração em tempo real, o que facilitou a participação simultânea da equipe no processo de design.

A ferramenta possibilitou a construção de protótipos navegáveis, permitindo visualizar como o usuário interage com o sistema antes da implementação. Também foram utilizados componentes reutilizáveis, garantindo padronização em elementos como botões, menus e formulários. O recurso de modo Dev contribuiu para a integração entre design e desenvolvimento, fornecendo informações detalhadas de espaçamento, cores e dimensões.

Com isso, o Figma teve um papel importante no projeto ao transformar os requisitos levantados em interfaces claras e funcionais, servindo de guia para a etapa de desenvolvimento e ajudando a alinhar o trabalho da equipe.

Arquitetura de alto nível (componentes, módulos).

## Critérios de Aceite e Testes

Estratégia de testes (unitário, integração, aceitação).

Matriz de rastreabilidade: requisitos ↔ casos de teste ↔ entregas.

## Riscos e Impactos

Principais riscos técnicos e de negócio.

Planos de mitigação.

## Anexos

Link no Trello:

Especificação de Requisitos:

[Especificação de Requisitos:](https://trello.com/c/pMqCKKQH/24-documento-de-requisitos-de-software-sukatech)

Arquitetura Técnica do Sistema

[Arquitetura Técnica do Sistema](https://trello.com/c/doo6iQUX/25-arquitetura-do-sistema-sukatech)

Backlog do Produto

<https://trello.com/c/Otl7Dz1A/12-backlog-do-produto>

Cronograma do Projeto Miro

[Cronograma do Projeto](https://miro.com/welcomeonboard/bWRSWUptbEJCYy9uNXFyeHdrZUVtN05uRC9ZQ0gzRCtyNDcxaitsd2g1dURHbFdoaHNIUFRMVjJzYnZUWnlvYjNxMC9kK0JaVGltRnkyRVRaVzBlT0tDRjRqc1BDVHB1cTNKQ3hObFBiL2dCNlFTMUZOQ3J4dHR6SDdYVjJ2ZFNBS2NFMDFkcUNFSnM0d3FEN050ekl3PT0hdjE=?share_link_id=67897822403)

Repositório do Projeto github

<https://github.com/BRISA-UFG>