

## Sprint Documentation

<b>Sprint 04</b>	<b>Start Date:07/11/2025</b>	<b>Final Date:11/11/2025</b>
<b>Projetc Name:</b>	<b>Civitas-frontend</b>	
<b>Ano: 4º</b>	<b>Análise e Desenvolvimento de Sistemas - AMS</b>	
<b>Team Members</b>		
Karen Valião de Abreu		
João Antônio Neves		

## Sprint Backlog

Task#	Description	Start Date	Final date
66	Desenvolver Página Listar Instituições	07/11/2025	11/11/2025

## Task Description

Task #	Description	Assigned To	Status	Estimated Hours	Logged Hours
66	Desenvolver a página de listagem de instituições, baseada no protótipo UX/UI aprovado, garantindo visualização clara, ações rápidas e consistência com o design system.	Karen Valião de Abreu e João Antônio Neves	Finished	10	10

## Sprint Description

Durante a Sprint 04, foi atribuída à equipe a tarefa de desenvolver a página de listagem de instituições, que tem como principal objetivo apresentar de forma organizada e intuitiva as informações das instituições cadastradas no sistema.

Essa tela permite ao usuário consultar, filtrar, visualizar e gerenciar os registros, tornando a navegação e manutenção mais eficiente.

O design da página foi construído com base no protótipo UX/UI aprovado no Figma, respeitando as diretrizes de identidade visual e o design system estabelecido pelo projeto.

Foram aplicados princípios de hierarquia visual, responsividade e usabilidade, assegurando uma experiência consistente em diferentes dispositivos (desktop, tablet e mobile).

A tarefa envolveu a implementação completa da interface, organização dos componentes reutilizáveis e estruturação da lógica de filtros e ações. O foco principal foi garantir a clareza na exibição dos dados e a fluidez nas interações do usuário com os elementos da interface.

Ao final da Sprint, a página foi validada quanto à fidelidade visual ao protótipo, coerência com as demais telas do sistema e pronta para integração com a API.

## Sprint Results

Durante o desenvolvimento da página Listar Instituições, foram implementadas todas as funcionalidades previstas, resultando em uma interface completa e funcional .

O código foi estruturado em um componente funcional React, contendo:

- Definição de uma interface TypeScript (`Instituicao`) para padronizar os dados exibidos na tabela.
- Criação de uma lista mock (`mockInstituicoes`) com objetos representando instituições (nome, CNPJ, endereço, telefone, e-mail e status).
- Implementação de estado local (`useState`) para armazenar e controlar os filtros ativos (nome, CNPJ, cidade, estado e situação).
- Criação das funções principais:
  - `handleFiltroChange`: atualiza dinamicamente os filtros conforme o usuário digita.
  - `limparFiltros`: reseta todos os filtros para os valores iniciais.
  - `filtrarInstituicoes`: prepara a lógica de filtragem (atualmente simulada via `console.log`, pronta para integração com o back-end).
  - `handleAcao`: gerencia ações nos botões de visualizar, editar, excluir e cadastrar.

A estrutura da interface foi dividida em seções bem definidas:

- Cabeçalho (Header): exibe o título “Listagem de Instituições”, breadcrumbs de navegação e o botão de voltar.
- Área de Filtros: seção destacada com fundo escuro (`bg-[#2E2E2E]`), inputs e selects estilizados com bordas arredondadas e realce na cor institucional (`#0D5660`).
- Tabela de Dados: montada com base no mock, exibindo cada instituição com colunas bem definidas. A situação é destacada visualmente com badges coloridos — verde para “Ativa” e vermelho para “Inativa”.
- Ações rápidas: ícones de visualização, edição e exclusão utilizando o Material Symbols, aplicando feedback visual no hover.
- Botão “+ Cadastrar”: implementado conforme o design do Figma, estilizado em tom verde-escuro e com efeito de hover.

Outros destaques técnicos:

- Utilização de grid responsivo (`grid-cols-1 sm:grid-cols-2 lg:grid-cols-6`) para organizar os filtros.
- Aplicação de componentização semântica e classes utilitárias do Tailwind, garantindo legibilidade e fácil manutenção.
- Adaptação completa do layout para diferentes resoluções, com comportamento fluido em telas pequenas e largas.
- Implementação de validação visual em interações (hover, focus, seleção).

A estrutura final do código garante fácil integração futura com o back-end, bastando substituir o mock de dados por uma chamada à API e aplicar o filtro diretamente sobre o retorno real.

## Resultado:



## Listagem de Instituições

Home < Instituições < Listagem

← Voltar

**Busca:**  
Aqui você busca e filtra

Nome

CNPJ

Cidade

Todos Estados

Todas Situações

+ Cadastrar

Filtrar

Limpar

ID	Situação	Nome	Razão Social	CNPJ	CEP	Logradouro	Número	Bairro	Cidade	Estado	Telefone	Email	Ações	
1	Ativa	Fatec São Paulo	Centro Estadual...	12.345.678/0001/90	01124-060	Av.Tiradentes	615	Luz	São Paulo	SP	(11) 80028-922	fatec@sp.gov.br		
2	Inativa	UNICAMP	Inst. Federal...	12.345.678/0001/90	01124-060	Av.Tiradentes	617	Butantã	São Paulo	SP	(11) 80048-922	unicamp@gov.br		
3	Ativa	Fatec Jales	Centro Estadual...	32.345.678/0001/90	01124-060	Av.Tirage	619	Centro	São Paulo	SP	(11) 80028-922	fatec@sp.gov.br		

## Assinaturas