

Criar projeto de Microserviços

Dentre as opções de projetos *low-code* disponíveis no Cronapp estão os projetos de Microserviços. Um microserviço é uma abordagem arquitetônica e organizacional do desenvolvimento de *software* na qual consiste em pequenos serviços independentes que se comunicam usando APIs bem definidas e, por isso, não possui os recursos de interfaces cliente de um projeto *web* ou *mobile*, mas apenas os recursos de uma aplicação servidor.

Dessa forma, projetos de Microserviços permitem a criação de um projeto simplificado, sem muitas funcionalidades predefinidas, oferecendo ao usuário autonomia para desenvolver os serviços necessários à sua aplicação com agilidade, baixo acoplamento, escalabilidade e flexibilidade. Nesta documentação, abordaremos as principais características de projetos de Microserviços, os recursos disponíveis e as diferenças entre os demais tipos de projetos *low-code* do Cronapp.

Criar projeto

Para criar um projeto do tipo Microserviços, clique no botão **Novo Projeto** (seta da Figura 1). A primeira janela de configuração será exibida, permitindo a escolha do tipo do projeto. Selecione o tipo **Microserviços** (destaque 1), em seguida, informe um nome (2) e escolha a opção **Baseado em Modelo existente** (3), por fim, clique em **Finalizar**. Para mais informações dos campos de configuração, consulte a documentação [Criar projeto](#).

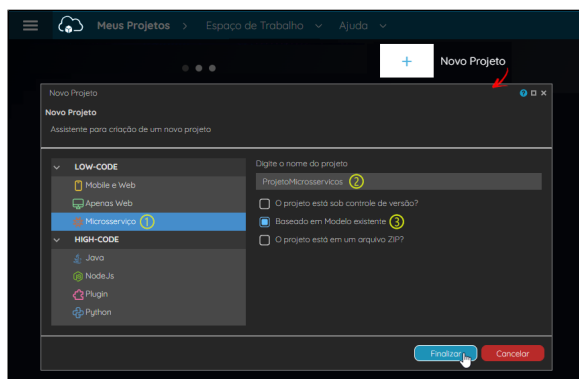


Figura 1 - Criação de um projeto de Microserviços no Cronapp

Modelos

Atualmente dois modelos de microserviços estão disponíveis: **Serviço de dados**, que oferecem serviços suportados por banco de dados, e **Serviço de negócio**, um modelo mais simplificado. Selecione a opção desejada e clique em **Avançar**. Abaixo, listamos as principais características de cada modelo.

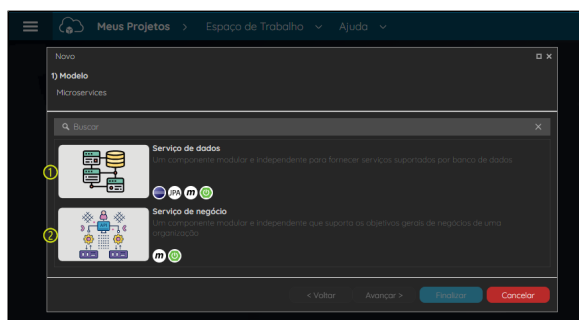


Figura 1.1 - Modelos disponíveis para o projeto de Microserviços

Nesta página

- [Criar projeto](#)
 - [Modelos](#)
 - [Configuração](#)
- [Recursos disponíveis](#)
 - [Árvore de arquivos](#)
 - [Menu Projetos](#)
 - [Ferramentas](#)
- [Recursos não disponíveis](#)
- [Autenticação](#)
- [Executando o projeto](#)
 - [Bloco de programação](#)
 - [Fonte de dados](#)

Saiba mais

- [Projeto de Microserviços com autenticação](#)

1. **Serviço de dados:** fornece ao usuário a funcionalidade de gerenciar serviços suportados por banco de dados. Esse modelo possui mais recursos previamente configurados em comparação com o de "Serviço de negócio", por exemplo, já possui uma classe "Sample" no [Diagrama](#) e a tabela correspondente no [Banco de dados](#), além do arquivo *persistence.xml*.
2. **Serviços de negócios:** define um projeto mais simplificado e fornece ao usuário a autonomia para desenvolver a aplicação de acordo com os objetivos gerais dos negócios de sua organização. Este modelo possui menos recursos configurados em comparação com o modelo anterior, por exemplo, não possui um [Diagrama](#) e [Banco de dados](#) configurados. No entanto, é possível configurar essas funcionalidades conforme a necessidade.

Configuração

A próxima tela permite definir as informações de configuração do **Id da Aplicação** e **Tipo de Autorização**. Essa mesma janela (Figura 1.2) será exibida, independente do modelo escolhido na janela anterior (Figura 1.1). Para concluir a criação do projeto, clique em **Finalizar**.

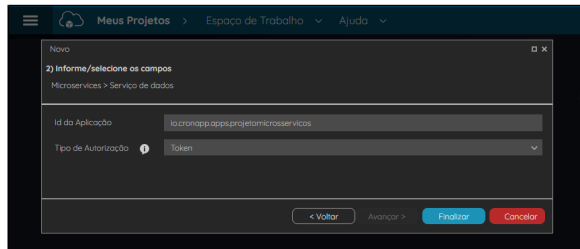


Figura 1.2 - Janela de configurações iniciais do projeto

- **Id da Aplicação:** identificação da aplicação.
- **Tipo de Autorização:** define o tipo de autorização dos serviços. No contexto de acesso a serviços, apenas o tipo "Token" está disponível, sendo empregado para conceder ou negar acesso a serviços específicos do sistema.

Ao finalizar as configurações de criação do projeto, o Cronapp exibirá uma pop-up perguntando se você deseja habilitar o [backup automático](#) desse projeto. Após a escolha ou não de *backup*, a criação do projeto será concluída.

Recursos disponíveis

Projetos de Microserviços apresentam menos funcionalidades quando comparados a projetos *web* ou *mobile*, uma vez que não incluem os recursos de interfaces cliente. Além disso, algumas funcionalidades possuem algumas diferenças em seus recursos. A seguir, destacamos os principais recursos e funcionalidades disponíveis nesse tipo de projeto no Cronapp.

Árvore de arquivos

A árvore de arquivos exibe os diretórios e arquivos pertencentes ao projeto. Abaixo listamos os diretórios disponíveis em projetos de Microserviços. Para detalhes sobre a estrutura de arquivos do projeto, acesse a documentação [Estrutura de arquivos](#).

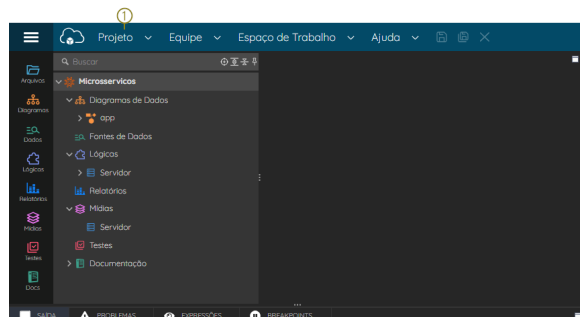


Figura 2 - Árvore de recursos de um projeto **Microserviços** baseado no modelo **Serviço de negócio**

- **Diagramas de Dados:** apenas os projetos criados utilizando o [modelo](#) "Serviço de dados", possuem o diagrama "app". No entanto, ao utilizar o modelo "Serviço de negócio", também é possível configurá-lo.
- **Fontes de Dados:** projetos de Microserviços não possuem Fontes de dados criadas previamente, dando autonomia ao usuário para criá-las se necessário.
- **Lógicas:** possui apenas blocos de programação servidor. Por padrão, já apresenta o [blockly](#) "HelloWorld".
- **Relatórios:** armazena os relatórios do projeto.
- **Mídias:** diretório virtual para agrupar arquivos estáticos do projeto.
- **Testes:** pasta reservada para os blocos de programação de teste do projeto.
- **Documentação:** diretório usado para manter os arquivos de documentação do projeto.

Menu Projetos

O menu de Sistemas **Projeto** (destaque 1 da figura 2) possui recursos para gerenciar e configurar o projeto. Abaixo listamos as principais funcionalidades presentes nos projetos de Microserviços.

- **Banco de dados:** apenas os projetos criados utilizando o [modelo](#) "Serviço de dados", possuem tabelas no banco de dados. No entanto, ao utilizar o modelo "Serviço de negócio", também é possível configurá-lo.
- **Chave de internacionalização:** está habilitada apenas na opção servidor.
- **Eventos e ações:** apenas o evento "Depois de Iniciar Sistema" está disponível em projetos de "Microserviços". Para mais informações desse evento, acesse a documentação [Eventos e ações](#).
- **Permissão de Segurança:** abre a tela para configuração apenas dos permissionáveis.
- **Gerenciamento de Tokens:** permite a configuração de tokens para autenticação por chave de API (*Api Key*) e *OAuth 2.0*.
- **Parâmetros do sistema:** abre a janela de parâmetros do sistema.
- **Agendador de tarefas:** abre a janela para criação de *jobs*.
- **Gerar documentação:** permite gerar documentações automáticas do sistema.
- **Configurações:** abre a janela de configurações do projeto. Alguns campos foram removidos nas abas "Geral", como os campos de "Rota Principal Web" e "Rota Principal Mobile", e "Configuração do Projeto", como os campos de "Tema" e "Autenticação".

Ferramentas

Em ferramentas é possível fazer o gerenciamento dos tipos de bancos de dados e dos backups automáticos, além de abrir o terminal. Projetos de Microserviços possuem as mesmas ferramentas que um projeto *web* ou *mobile*: [Gerenciador de SQL](#), [Terminal](#), [Testador de consultas JPQL](#) e [Gerenciamento de Backups](#).

Recursos não disponíveis

Visto que seu principal objetivo é apenas gerenciar e disponibilizar serviços do lado servidor, não é necessário o uso de interfaces cliente. Desse modo, projetos de **Microserviços** não possuem os recursos de [Fragmentos](#), [Formulários](#), [Dashboards](#), [Skins de Tema](#) e [Fluxo de Trabalho](#).

Autenticação

Projetos de Microserviços não possuem recursos de autenticação próprios. Se houver a necessidade de disponibilizar um serviço de forma privada, exigindo a autenticação de usuários, será necessário integrar outra aplicação para lidar com a autenticação. Para mais informações de integração com outras aplicações, consulte o tutorial [Projeto de Microserviços com autenticação](#). Também é importante destacar que os **Permissionáveis** do projeto de Microserviços devem ser os mesmos disponíveis no projeto de autenticação. Para configurar os permissionáveis, acesse a ferramenta [Permissão de Segurança](#) (Figura 3). As alterações realizadas nessa ferramenta irá afetar o arquivo `authorities.json` (Endereço: `src/main/java/META-INF/authorities.json`).

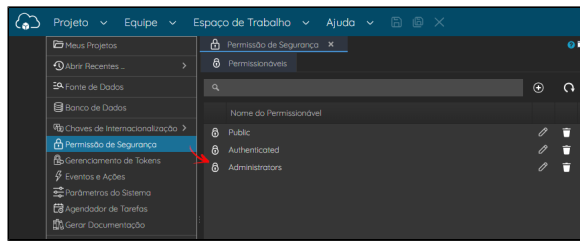


Figura 3 - Ferramenta Permissão de Segurança

Executando o projeto

Visto que projetos de Microsserviços são do tipo servidor, seus recursos serão executados a partir de requisições REST. Dessa forma, estão disponíveis apenas duas opções de execução do projeto, conforme apresentadas na imagem abaixo. Para mais informações, acesse a documentação do [Swagger](#) - [OpenAPI](#).

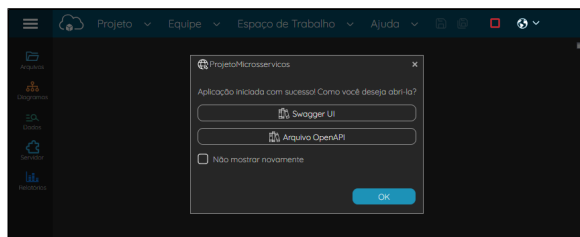


Figura 4 - Executando o projeto

- **Swagger UI**: abre outra aba no navegador com a página do [Swagger](#).
- **Arquivo OpenAPI**: abre outra aba no navegador com o esquema do OpenAPI.

Bloco de programação

Na imagem abaixo, podemos visualizar o *blockly* "HelloWorld" e a função "Executar" que já estão configurados em projetos de Microsserviços. Essa função, assim como as demais criadas, serão executadas a partir de requisições REST. Para configurar o endereço REST de uma função de bloco de programação, clique no ícone de engrenagem (destaque 1 da Figura 5) do editor de *blockly*. Para mais informações de configuração, acesse o tópico "Propriedades do Blockly" da documentação [Bloco de programação](#).



Figura 5 - Bloco de programação HelloWorld

Na imagem a seguir, podemos observar a execução da função da Figura 5 no [Swagger](#). Nessa página, podemos visualizar o *endpoint* (destaque 1 da Figura 5.1) e o resultado de execução da função (destaque 2). Ao executar em ambientes externos, é necessário concatenar o *endpoint* da função com o domínio da aplicação.

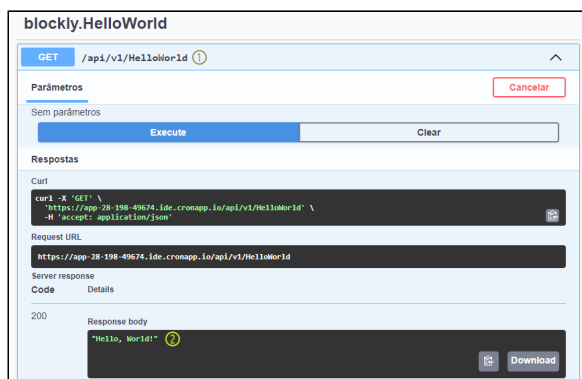


Figura 5.1 - Execução da função da Figura

Fonte de dados

Projetos de Microsserviços não possuem [Fonte de dados](#) criadas previamente, oferecendo aos usuários autonomia para criá-las conforme a necessidade. Para o exemplo da imagem abaixo, criamos uma classe "Usuarios" no [diagrama de dados](#) e, a partir dela, a [Fonte de dados](#) "Usuarios" (destaque 1 da Figura 5.2). Em seguida, na aba [Ações](#) (2), marcamos a [checkbox](#) da ação "Para Inserir" (verbo HTTP POST) e a deixamos como pública. Também habilitamos a opção "Swagger/Open API" (3). Para mais informações de configuração, acesse o tópico "Como configurar" da documentação do [Swagger](#).

Para executar as ações da [Fonte de dados](#) em ambientes externos, o endereço REST (4) deverá ser concatenado com o domínio da aplicação. Na Figura 5.3, é possível visualizar a execução na página do [Swagger](#).

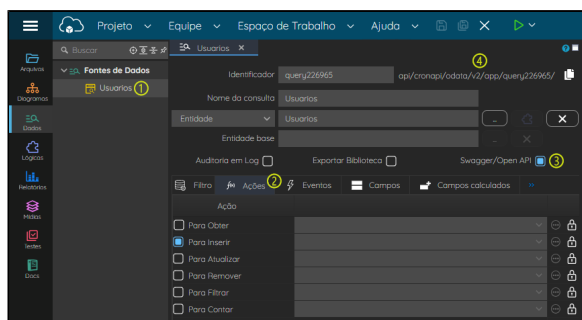


Figura 5.2 - Criação de uma Fonte de dados

A imagem a seguir mostra a execução do método POST da [Fonte de dados](#). No corpo da requisição, informamos os dados do usuário (destaque 1 da Figura 5.3) e, após a execução, esses dados (2) serão armazenados no banco de dados do sistema.

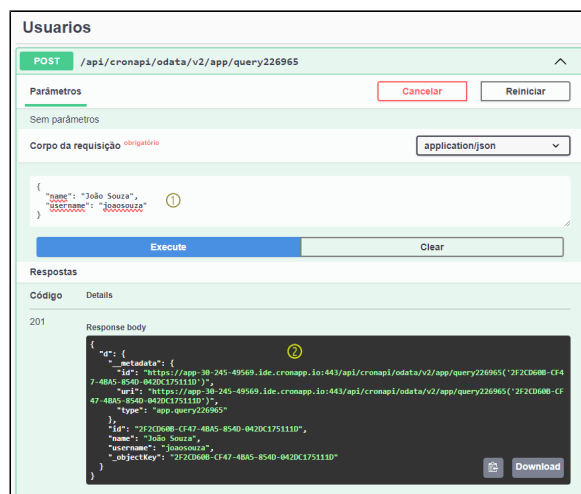


Figura 5.3 - Execução da requisição da Fonte de dados no Swagger