



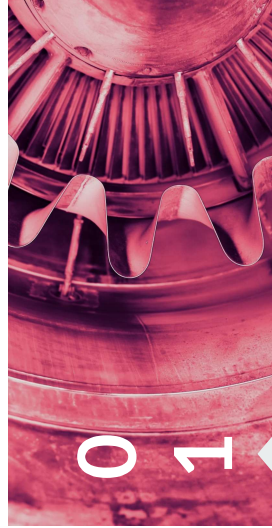
# Projeto BeeHus

Bom Young Kim



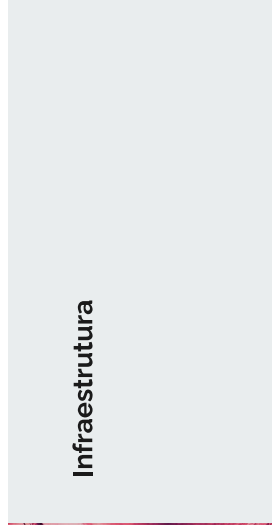


# Índice



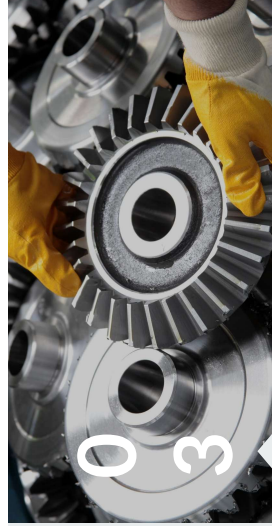
01

Visão Geral



Infraestrutura

02



03

Por que?



# Visão geral

A Beehus Tecnologia (Beehus Tech) é uma startup do setor de Wealth Tech, focada em desenvolver soluções tecnológicas para a consolidação e gestão de patrimônio de clientes no Brasil e no exterior. Possui como pilar fundamental, a tecnologia de dados. Utiliza múltiplas tecnologias para captura de dados, processamento diário, precificação de ativos e geração de relatórios personalizados para seus clientes. **O principal diferencial da Beehus é oferecer uma visão unificada e automática de todo o patrimônio.** A plataforma centraliza informações de diversos bancos e corretoras, automatizando processos manuais para dar ao gestor financeiro uma visão clara e 360 graus dos ativos, facilitando a tomada de decisões estratégicas de forma eficiente e segura.





# Infraestrutura

A combinação de Django, Celery, Celery Beat, e Redis forma uma arquitetura robusta para a construção de aplicações web modernas e escaláveis, que gerenciam tanto requisições síncronas quanto tarefas assíncronas complexas.

## Django

Principal (backend).  
Gerencia a interface do usuário, a lógica de negócios síncrona, banco de dados (ORM), autenticação e a API que aciona as tarefas.

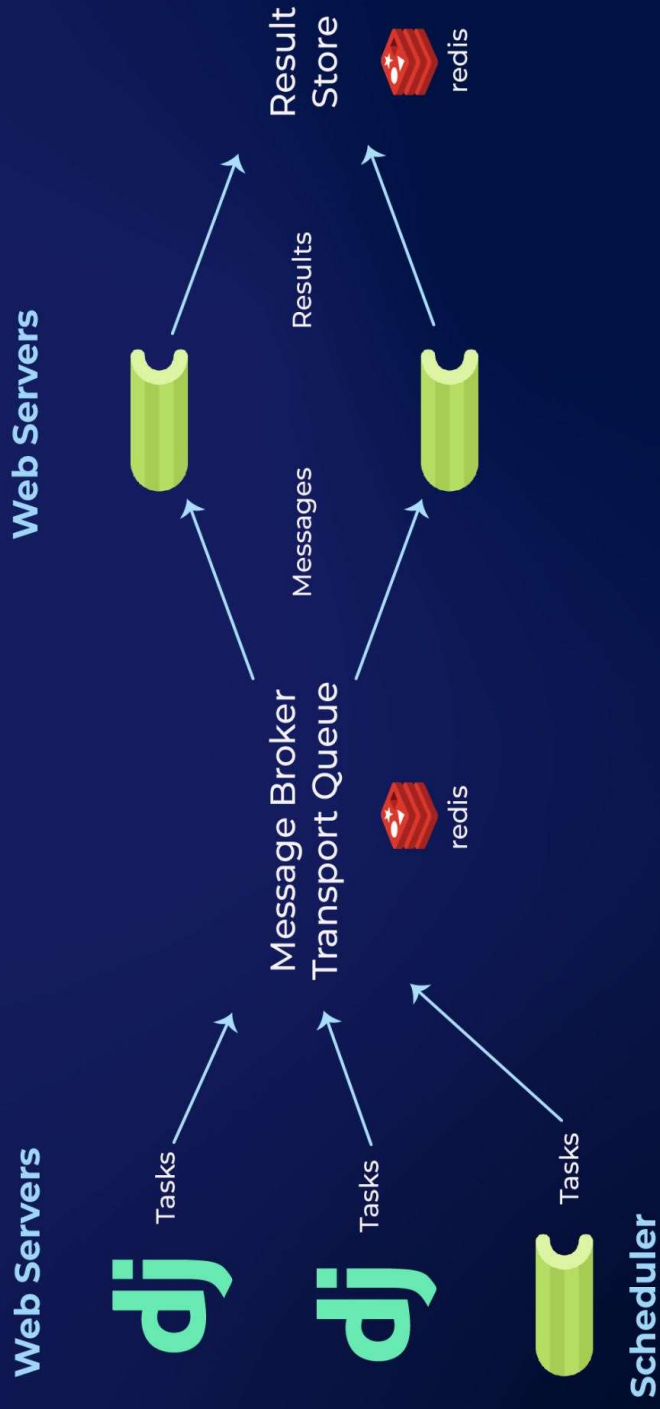
## Redis

Serve como a "ponte de comunicação" de alta velocidade, armazenando as tarefas a serem executadas (fila) e os resultados das tarefas concluídas.

## Celery

Executa tarefas demoradas (como extração de dados) em segundo plano, sem travar a resposta da aplicação web principal

# Architecture Of Celery





# Por que essa solução?

- 1**

Escalabilidade: Permite que o site continue rápido, mesmo com muitas tarefas. Você pode adicionar mais "trabalhadores" (workers) para dar conta do recado sem sobrecarregar a página principal.
- 2**

Rastreabilidade: Você consegue monitorar cada tarefa, sabendo se ela foi concluída ou se falhou. Ferramentas visuais mostram em tempo real o que está acontecendo nos bastidores.
- 3**

Eficiência: Tira tarefas pesadas (extração de dados em vários pontos) da frente do usuário. O aplicativo mais leve e os recursos do servidor são usados de forma inteligente e otimizada.
- 4**

Independência (com Docker): Cada parte do sistema (site, banco de dados, tarefas) funciona isolada em seu próprio "pacote". Isso facilita a instalação e evita problemas de compatibilidade.

Agendamento de tarefas.

Diversos cliente e fontes, vai necessitar de uma solução para minimizar falhas de gestão de informação.

Django administration
WELCOME ADMIN VIEW SITE / CHANGE PASSWORD / LOG OUT

Home Periodic Tasks Add periodic task

AUTHENTICATION AND AUTHORIZATION	
Groups	<a href="#">Add</a>
Users	<a href="#">Add</a>

PERIODIC TASKS	
Clocked	<a href="#">Add</a>
Crontabs	<a href="#">Add</a>
Intervals	<a href="#">Add</a>
Periodic tasks	<a href="#">Add</a>
Solar events	<a href="#">Add</a>

### Add periodic task

Name: Print time every 30 seconds ⓘ

Short Description For This Task

---

Task (registered): print\_time

Task (custom):

☒ Enabled  
Set to False to disable the schedule

Description:

Print the current time every 30 seconds

Detailed description about the details of this Periodic Task

### Schedule

Interval Schedule:  
every 30 seconds ↕ +  
Interval Schedule to run the task on. Set only one schedule type, leave the others null.

Crontab Schedule:  
----- ↕ +  
Crontab Schedule to run the task on. Set only one schedule type, leave the others null.

Solar Schedule:  
----- ↕ +  
Solar Schedule to run the task on. Set only one schedule type, leave the others null.





**Obrigado!**

