

**BOSCH**

Ana Luiza Bueno

Daniel de Souza

Davi Lima

Gabriel Jordão

Igor Bedon de Sousa

Marcelo Rodrigues

Vinícius Lira

## **Documentação de Testes**


Arquivo para documentação e planejamento de testes do Sistema de Inteligência Analítica para Projeção de Desempenho de Produtos para o MA/MBL-LA em conjunto com CaP/ETS.

**CAMPINAS-SP, BRASIL**

2024

## Sumário

1	Introdução .....	3
2	Escopo do Teste .....	3
2.1	Testes atendidos.....	3
3	Estratégia de Teste .....	4
3.1	Abordagens de Teste .....	4
3.1.1	Detalhamento das Estratégias .....	5
4	Ambiente de Teste .....	6
5	Critérios de Aceitação.....	7
5.1	Critérios Gerais.....	7
5.2	Critérios de Aceitação para Testes Rejeitados .....	8
6	Riscos .....	8
6.1	Riscos Técnicos .....	8
6.2	Riscos Humanos .....	9
6.3	Riscos Operacionais .....	9
7	Cronograma .....	10
8	Casos de Teste.....	11
9	Relatório de Bugs.....	16
10	Conclusão.....	18
11	Histórico de Revisões.....	18

 <b>BOSCH</b>	<b>PROJETO SIRIUS</b>	Versão 1.0	BOSCH MA/MBL-LA e CaP/ETS
------------------------------------------------------------------------------------------------	---------------------------	------------	------------------------------

## 1 Introdução

O projeto **Sirius** é um sistema com o objetivo de otimizar a gestão de portfólios de produtos do MA/MBL-LA. Sua principal funcionalidade é fornecer sugestões de itens que devem ser mantidos ou removidos do portfólio, utilizando análises baseadas em dados históricos e machine learning.

O objetivo principal dos testes descritos neste plano é assegurar que todas as funcionalidades do sistema Sirius operem corretamente, identificando e corrigindo falhas que possam comprometer a experiência do usuário ou os resultados gerados pelo sistema.

## 2 Escopo do Teste

O plano de testes para o sistema Sirius abrangerá as funcionalidades identificadas na análise de requisitos do projeto. Abaixo, detalha-se o que será testado e o que está fora do escopo.

### 2.1 Testes atendidos

- **Análise de dados em grande quantidade**

Validação da capacidade de processar e apresentar gráficos descritivos.

- **Visualização gráfica clara dos dados**


Testes relacionados à apresentação das informações e facilidade de interpretação pelos usuários.

- **Geração de previsões utilizando Machine Learning**

Validação da precisão e confiabilidade das previsões baseadas nos dados fornecidos.

- **Acessibilidade e segurança do sistema**

Testes de login, autorização de usuários e restrição de acesso.

 <b>BOSCH</b>	<b>PROJETO SIRIUS</b>	Versão 1.0	BOSCH MA/MBL-LA e CaP/ETS
------------------------------------------------------------------------------------------------	---------------------------	------------	------------------------------

- **Desempenho da plataforma**


Validação de tempo de resposta e estabilidade do sistema, considerando a quantidade de dados.

### 3 Estratégia de Teste

A abordagem será baseada em diferentes níveis e tipos de teste, conforme descrito abaixo.

#### 3.1 Abordagens de Teste

Funcionalidade	Tipo de Teste	Abordagem	Objetivo
Análise de dados em grande quantidade	Teste de Integração	Caixa-preta	Garantir que os dados sejam processados e apresentados corretamente.
Visualização gráfica clara dos dados	Teste de Interface	Caixa-preta	Validar a clareza e a usabilidade da interface gráfica para exibir informações.
Geração de previsões utilizando Machine Learning	Teste de Unidade	Caixa-branca	Testar os algoritmos de ML para verificar a precisão e confiabilidade das previsões.
Segurança e acesso restrito	Teste Funcional	Caixa-preta	Certificar-se de que apenas usuários autorizados podem acessar a plataforma.
Desempenho da plataforma	Teste de Desempenho	Caixa-preta	Avaliar o tempo de resposta e a estabilidade ao processar grandes volumes de dados.

 <b>BOSCH</b>	<b>PROJETO SIRIUS</b>	Versão 1.0	BOSCH MA/MBL-LA e CaP/ETS
------------------------------------------------------------------------------------------------	---------------------------	------------	------------------------------

### 3.1.1 Detalhamento das Estratégias

- **Testes de Unidade**

**Objetivo:** Validar componentes isolados, como algoritmos de Machine Learning e cálculos de KPI.

**Ferramentas:** Simulação de dados mockados para verificar precisão.

- **Testes de Integração**

**Objetivo:** Garantir que os módulos do sistema funcionem bem juntos, como o processamento de dados e a geração de gráficos.

**Ferramentas:** Testes realizados diretamente no ambiente de desenvolvimento.

- **Testes de Interface**

**Objetivo:** Verificar a usabilidade e clareza da interface para os usuários.

**Critérios:** A interface deve ser intuitiva, com gráficos claros e navegabilidade eficiente.

- **Testes Funcionais**

**Objetivo:** Validar funcionalidades descritas no escopo, como a restrição de acesso e geração de previsões.

**Critérios:** Cada funcionalidade deve operar conforme os requisitos especificados.

- **Testes de Desempenho**


**Objetivo:** Avaliar a capacidade do sistema de lidar com grandes volumes de dados sem comprometer o tempo de resposta.

**Cenários:** Carga normal e situações de pico.

- **Testes de Segurança**

**Objetivo:** Garantir que usuários autorizados consigam acessar o sistema.

**Métodos:** Tentativas de login não autorizado e avaliação de vulnerabilidades.

 <b>BOSCH</b>	<b>PROJETO SIRIUS</b>	Versão 1.0	BOSCH MA/MBL-LA e CaP/ETS
------------------------------------------------------------------------------------------------	---------------------------	------------	------------------------------

## 4 Ambiente de Teste

O ambiente de teste do sistema Sirius foi projetado para simular as condições reais de uso. Abaixo, descrevemos os componentes de hardware e software usados para a execução dos testes.

HARDWARE	
Componente	Especificação
Processador	11th Gen Intel(R) Core(TM) i5-1145G7 @ 2.60GHz
Memória RAM	16 GB
Armazenamento	256 GB SSD
Rede	Conexão estável de 100 Mbps ou superior

SOFTWARE	
Componente	Versão/Especificação
Sistema Operacional	Windows 11 / Linux Ubuntu 22.04
Navegador Web	Google Chrome
Ferramentas de Teste	Swagger (OpenAPI)
Banco de Dados	PostgreSQL
Ambiente de Desenvolvimento	Visual Studio Code, Docker


### Servidor de Teste:

O sistema será implantado em um ambiente de teste isolado, utilizando Docker para simular o ambiente de produção.

O banco de dados será colocado em um contêiner no Docker, replicando a estrutura utilizada na aplicação real.

### Mock de Dados:

Para validar as funcionalidades do sistema, será utilizado um conjunto de dados mockados representando as condições reais de uso.

 <b>BOSCH</b>	<b>PROJETO SIRIUS</b>	Versão 1.0	BOSCH MA/MBL-LA e CaP/ETS
------------------------------------------------------------------------------------------------	---------------------------	------------	------------------------------

Os dados mockados serão inseridos via Script Python no banco de dados antes dos testes.

#### **Simulação de Dispositivos:**

O sistema será testado em um PC com as especificações descritas.

Não há necessidade de testes em dispositivos móveis, uma vez que o sistema é exclusivamente para uso interno via web.

#### **Logs e Monitoramento:**

Todos os testes serão monitorados pelo serviço de logging do BetterStack

Esse ambiente de teste garante que todas as funcionalidades descritas no escopo do sistema Sirius sejam validadas em condições controladas, simulando o ambiente de produção.

## **5 Critérios de Aceitação**

Um teste será considerado bem-sucedido se atender aos seguintes critérios:

### **5.1 Critérios Gerais**

- **Funcionalidade:**


O sistema deve permitir a análise descritiva e preditiva do desempenho dos produtos no portfólio, gerando dashboards claros e compreensíveis.

Os gráficos e relatórios devem estar em conformidade com os dados fornecidos.

- **Usabilidade:**

A interface deve ser intuitiva, com uma curva de aprendizado mínima para os usuários internos da Bosch.

Todas as funcionalidades devem ser acessíveis em até 3 cliques.

 <b>BOSCH</b>	<b>PROJETO SIRIUS</b>	Versão 1.0	BOSCH MA/MBL-LA e CaP/ETS
------------------------------------------------------------------------------------------------	---------------------------	------------	------------------------------

#### **Performance:**

O tempo de carregamento das páginas deve ser inferior a 5 segundos.

As requisições feitas pela API, em relação a busca de informações de todos os produtos, devem demorar menos do que 10 segundos.

#### **Segurança:**

Somente usuários autorizados devem conseguir acessar a plataforma, conforme os requisitos de autenticação.

O sistema não deve permitir acessos não autorizados ou manipulação indevida de dados.

#### **Conformidade:**

O sistema deve estar em conformidade com os padrões e políticas da Bosch, incluindo confidencialidade e integridade dos dados.

## **5.2 Critérios de Aceitação para Testes Rejeitados**

Bugs encontrados durante os testes devem ser categorizados e documentados no Relatório de Bugs, destacando a gravidade, o impacto e as etapas para reprodução.

A correção dos bugs críticos é obrigatória antes do encerramento da fase de validação.

## **6 Riscos**


Os seguintes riscos podem impactar a execução e validação dos testes para o sistema **Sirius**:

### **6.1 Riscos Técnicos**

- **Acesso Limitado aos Dados Reais:**

Os dados utilizados são mockados, o que pode não refletir completamente as condições reais.



 <b>BOSCH</b>	<b>PROJETO SIRIUS</b>	Versão 1.0	BOSCH MA/MBL-LA e CaP/ETS
------------------------------------------------------------------------------------------------	---------------------------	------------	------------------------------

**Impacto:** Redução na assertividade das análises preditivas.

- **Compatibilidade de Software:**

Incompatibilidades com ferramentas, como por exemplo o Docker, podem atrasar os testes.

**Mitigação:** Garantir que todas as ferramentas estejam atualizadas e configuradas previamente.

- **Curto Prazo de Desenvolvimento:**

O cronograma curto pode comprometer a cobertura total dos testes.

**Mitigação:** Priorizar testes críticos e adotar estratégias de teste automatizado para otimizar tempo.

## 6.2 Riscos Humanos

- **Falta de Conhecimento do Testador:**

Membros da equipe de teste podem não estar familiarizados com as ferramentas ou o sistema.

**Mitigação:** Realizar cursos curtos e buscar documentação detalhada.

- **Erros de Comunicação:**

A ausência de clareza na troca de informações entre testadores e desenvolvedores pode levar à identificação inadequada de bugs.


**Mitigação:** Utilização do Jira e contato frequente para ajustar a comunicação.

## 6.3 Riscos Operacionais

- **Dependência de Ferramentas Externas:**

Ferramentas externas, como Docker ou o mock de dados, podem apresentar falhas técnicas.


**Mitigação:** Ter backups e ambientes alternativos para continuidade dos testes.

 <b>BOSCH</b>	<b>PROJETO SIRIUS</b>	Versão 1.0	BOSCH MA/MBL-LA e CaP/ETS
------------------------------------------------------------------------------------------------	---------------------------	------------	------------------------------

## 7 Cronograma

O cronograma para a realização dos testes do sistema Sirius foi elaborado considerando o escopo, os tipos de testes propostos, e os riscos identificados.

<b>Etapas</b>	<b>Descrição</b>	<b>Responsável</b>	<b>Duração Estimada</b>	<b>Data de Início</b>	<b>Data de Término</b>
Configuração do Ambiente	Configuração do ambiente de teste	Equipe BackEnd	2 dias	04/11/2024	05/11/2024
Testes de Unidade	Testes individuais para cada funcionalidade	Desenvolvedores	4 dias	05/11/2024	08/11/2024
Testes de Integração	Validação da interação entre os módulos do sistema.	Desenvolvedores	3 dias	11/11/2024	13/11/2024
Testes de Interface (UI/UX)	Avaliação da usabilidade e acessibilidade	Equipe FrontEnd	3 dias	13/11/2024	15/11/2024
Testes Funcionais	Verificação do cumprimento dos requisitos funcionais.	Desenvolvedores	3 dias	18/11/2024	20/11/2024
Testes de Desempenho	Medição do tempo de resposta e carga do sistema	Desenvolvedores	1 dia	21/11/2024	21/11/2024
Testes de Segurança	Validação da autenticação, autorização e proteção de dados	Desenvolvedores	1 dia	22/11/2024	22/11/2024
Testes de Aceitação	Execução de casos de teste	MA/MLA-LA	1 dia	22/11/2024	22/11/2024

 <b>BOSCH</b>	<b>PROJETO SIRIUS</b>	Versão 1.0	BOSCH MA/MBL-LA e CaP/ETS
------------------------------------------------------------------------------------------------	---------------------------	------------	------------------------------

	pelos stakeholders				
Correção de Bugs	Revisão e correção de falhas	Desenvolvedores	3 dias	25/11/2024	27/11/2024
Validação Final e Documentação	Revisão final e compilação do relatório de testes	MA/MBL-LA	1 dia	27/11/2024	27/11/2024

**Flexibilidade:** O cronograma pode ser ajustado conforme a complexidade dos bugs encontrados durante os testes.

**Paralelismo:** Algumas etapas, principalmente as desempenhadas por equipe diferentes, podem ser realizadas em paralelo.

## 8 Casos de Teste

### Caso de Teste 1: Geração de Gráficos Descritivos

**ID:** CT001

**Título:** Gerar gráficos descritivos com base em dados mock

**Objetivo do Teste:** Verificar se o sistema gera gráficos descritivos corretamente a partir dos dados mock.

#### Pré-condições (Given):


O usuário está autenticado na plataforma Sirius.

Os dados mock estão disponíveis e carregados no sistema.

#### Passos do Teste (When):

O usuário acessa a funcionalidade "Portfólio".

O usuário seleciona um produto específico do portfólio.

 <b>BOSCH</b>	<b>PROJETO SIRIUS</b>	Versão 1.0	BOSCH MA/MBL-LA e CaP/ETS
------------------------------------------------------------------------------------------------	---------------------------	------------	------------------------------

**Resultado Esperado (Then):**

O sistema exibe um gráfico descritivo correto com base nos dados fornecidos.

As informações são apresentadas de forma clara e compreensível.

**Dados de Teste:**

Produto ID: 0986B02320

**Tipo de Teste:**

Teste Funcional

**Autor do Caso de Teste:**

Desenvolvedores

**Data de Criação:**

01/11/2024

**Status:**

Executado

**Caso de Teste 2:** Previsão de Desempenho com IA

**ID:** CT002

**Título:** Validar previsão de desempenho gerada pela IA

**Objetivo do Teste:** Validar se a previsão de desempenho gerada pela IA está correta com base nos dados mock.


**Pré-condições (Given):**

O usuário está autenticado na plataforma Sirius.

Os dados mock estão disponíveis no sistema.

**Passos do Teste (When):**

O usuário seleciona um produto no portfólio.

 <b>BOSCH</b>	<b>PROJETO SIRIUS</b>	Versão 1.0	BOSCH MA/MBL-LA e CaP/ETS
------------------------------------------------------------------------------------------------	---------------------------	------------	------------------------------

O usuário solicita a previsão de desempenho para o produto.

**Resultado Esperado (Then):**

O sistema retorna uma previsão clara gerada pela IA.

O resultado é exibido no gráfico.

**Dados de Teste:**

Produto ID: 0986B02320

**Tipo de Teste:**

Teste Funcional

**Autor do Caso de Teste:**

Desenvolvedores

**Data de Criação:**

01/11/2024

**Status:**

Executado

**Caso de Teste 3: Interface Gráfica (UI/UX)**

**ID:** CT003

**Título:** Avaliar responsividade e intuitividade da interface gráfica


**Objetivo do Teste:** Garantir que a interface gráfica é responsiva e intuitiva para navegação.

**Pré-condições (Given):**

O usuário acessa a página inicial da plataforma Sirius.

**Passos do Teste (When):**

O usuário navega pelas funcionalidades da plataforma.

 <b>BOSCH</b>	<b>PROJETO SIRIUS</b>	Versão 1.0	BOSCH MA/MBL-LA e CaP/ETS
------------------------------------------------------------------------------------------------	---------------------------	------------	------------------------------

O usuário interage com elementos gráficos, como botões e gráficos.

**Resultado Esperado (Then):**

A interface é responsiva e apresenta os elementos de forma clara e organizada.

**Tipo de Teste:**

Teste Funcional

**Autor do Caso de Teste:**

Desenvolvedores

**Data de Criação:**

01/11/2024

**Status:**

Executado

**Caso de Teste 4:** Autenticação de Usuário

**ID:** CT004

**Título:** Garantir acesso apenas a usuários autorizados

**Objetivo do Teste:** Validar que apenas usuários autorizados conseguem acessar a plataforma.


**Pré-condições (Given):**

O sistema confere as credenciais válidas na Azure.

**Passos do Teste (When):**

O usuário insere as credenciais na página de login.

O sistema valida as credenciais na Azure.

 <b>BOSCH</b>	<b>PROJETO SIRIUS</b>	Versão 1.0	BOSCH MA/MBL-LA e CaP/ETS
------------------------------------------------------------------------------------------------	---------------------------	------------	------------------------------

**Resultado Esperado (Then):**

O sistema autentica usuários autorizados com sucesso.

O usuário é redirecionado para a página inicial.

**Tipo de Teste:**

Teste de Segurança

**Autor do Caso de Teste:**

Desenvolvedores

**Data de Criação:**

01/11/2024

**Status:**

Executado

**Caso de Teste 5:** Performance de Geração de Gráficos Gerais

**ID:** CT005

**Título:** Validar o tempo de resposta da geração de gráficos que envolvem todos os produtos

**Objetivo do Teste:** Verificar se o gráfico é gerado em até 10 segundos.

**Pré-condições (Given):**

O usuário acessa a funcionalidade "Análise Geral".


Os dados mock estão carregados no sistema.

**Passos do Teste (When):**

O usuário seleciona um o gráfico que deseja ver na Análise Geral.

O usuário solicita a geração do gráfico.

**Resultado Esperado (Then):**

 <b>BOSCH</b>	<b>PROJETO SIRIUS</b>	Versão 1.0	BOSCH MA/MBL-LA e CaP/ETS
------------------------------------------------------------------------------------------------	---------------------------	------------	------------------------------

O gráfico é gerado em no máximo 10 segundos.

O sistema não apresenta erros ou travamentos.

**Tipo de Teste:**

Teste de Performance

**Autor do Caso de Teste:**

Desenvolvedores

**Data de Criação:**

01/11/2024

**Status:**

Executado

## 9 Relatório de Bugs

**ID do Bug:** BUG001

**Título do Bug:** Desempenho ruim ao carregar informações de todos os produtos de uma vez.

**Descrição:**

Ao fazermos requisições que precisavam trazer os dados de todos os produtos de uma vez, como a da funcionalidade “Análise Geral”, não havia retorno devido a quantidade excessiva de dados a serem passados na resposta, mesmo com a paginação.

**Informações do Bug:**

**Gravidade:** Média

**Prioridade:** Alta

**Status:** Corrigido

**Data de Detecção:** 21/11/2024



### Pré-condições:

Usuário autenticado na plataforma Sirius.

Dados mock carregados.

### Passos:

Acesse a funcionalidade “Análise Geral”.

Selecione o gráfico que deseja visualizar.

### Resultado Esperado:

O gráfico deve ser totalmente carregado em menos de 10 segundos.

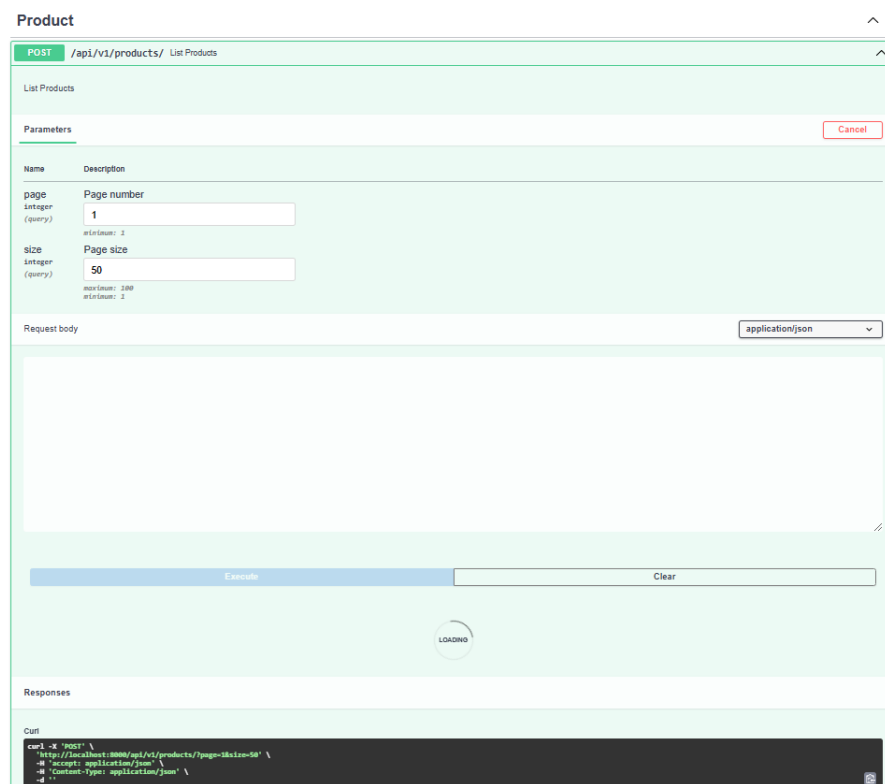
### Resultado Real:

O gráfico não era renderizado

### Anexos:

#### Sirius Service - Info Provider

[/openapi.json](#)



Product

POST /api/v1/products/ List Products

List Products

Parameters

Name	Description
page integer (query)	Page number 1 minimum: 1
size integer (query)	Page size 50 minimum: 100 maximum: 2

Request body


application/json

Execute Clear

LOADING

Responses

```
curl -X 'POST' \
  'http://localhost:8080/api/v1/products/?page=1&size=50' \
  -H 'accept: application/json' \
  -H 'Content-Type: application/json' \
  -d ''
```

 <b>BOSCH</b>	<b>PROJETO SIRIUS</b>	Versão 1.0	BOSCH MA/MBL-LA e CaP/ETS
------------------------------------------------------------------------------------------------	---------------------------	------------	------------------------------

**Autor do Relatório:** Equipe FrontEnd

**Data do Relatório:** 22/11/2024

**Responsável pela Correção:** Equipe BackEnd

## 10 Conclusão

Os testes realizados no sistema Sirius permitiram validar as funcionalidades descritas no escopo do projeto, garantindo que o sistema atende aos principais requisitos para a análise preditiva e descritiva de produtos do portfólio.

Além disso, a execução dos testes evidenciou a necessidade de aprimoramentos futuros, como a integração com dados em tempo real e a maior cobertura dos cenários de teste para situações limite.

Em suma, o sistema Sirius apresenta um nível satisfatório de qualidade, estando apto para ser entregue como uma prova de conceito (PoC) funcional, alinhada aos objetivos estabelecidos pela equipe de desenvolvimento e pelos stakeholders.

## 11 Histórico de Revisões

Versão	Data	Autor(es)	Descrição
0.1	04/11/2024	Igor Bedon de Sousa	Criação do documento inicial
1.0	23/11/2024	Igor Bedon de Sousa	Finalização e Atualização