

# **Plano de Testes**

Desenvolvimento de Sistemas

**Professor: Daricélio Moreira Soares**

**Equipe: Cleyciane Farias de Lima  
Juliana Abreu da Cunha,  
Raul Vitor Lopes da Costa**

**Junho / 2019**

## Histórico de Revisões

<b>Data</b>	<b>Versão</b>	<b>Descrição</b>	<b>Autor</b>
11/06/2019	1.0	Criação do Documento.	Raul Vitor Lopes da Costa.
12/06/2019	1.1	Tipos de Teste para o Sistema Incluídos.	Raul Vitor Lopes da Costa.
12/06/2019	1.2	Tipos de Requisitos a serem testados e Incluídos.	Juliana Abreu da Cunha
12/06/2019	1.3	O escopo, identificação do projeto e os principais objetivos do sistema	Cleyciane Farias de Lima
12/06/2019	1.4	Inclusão dos Recursos e Cronograma.	Cleyciane Farias de Lima, Raul Vitor Lopes da Costa.

# Índice

<b>1. INTRODUÇÃO .....</b>	<b>4</b>
1.1 OBJETIVOS.....	4
1.2 O CRONOS .....	4
1.3 ESCOPO .....	4
<b>2. REQUISITOS A TESTAR.....</b>	<b>5</b>
2.1 TESTE DO INTEGRIDADE DE DADOS .....	5
2.2 TESTE FUNCIONAL.....	5
2.3 TESTE DA INTERFACE DO USUÁRIO .....	5
2.4 PERFIL DA PERFORMANCE .....	5
2.5 TESTE DE INSTALAÇÃO .....	5
<b>3. ESTRATÉGIA DE TESTE.....</b>	<b>6</b>
3.1 TIPOS DE TESTE .....	6
3.1.1 <i>Teste de Integridade de Dados</i> .....	6
3.1.2 <i>Teste de Função</i> .....	7
3.1.3 <i>Teste da Interface do Usuário</i> .....	7
3.1.4 <i>Teste de Performance</i> .....	8
3.1.5 <i>Teste de Instalação</i> .....	9
<b>4. RECURSOS .....</b>	<b>9</b>
4.1 TRABALHADORES .....	10
4.2 SISTEMA .....	11
<b>5. CRONOGRAMA.....</b>	<b>12</b>

## **1. Introdução**

Este documento de Plano de Teste tem o objetivo de documentar as informações necessárias para planejar e controlar os testes de validação do projeto CRONOS. O documento descreve o plano geral de testes referente as verificações da base de dados em arquivos CSV a serem executados para validar o produto.

### **1.1 Objetivos**

Este documento do Plano de Testes do sistema CRONOS, constitui-se dos seguintes objetivos: identificar a disponibilidade dos projetos existentes, os requisitos de software que devem ser testados, recomendar descrever os testes que deverão ser utilizadas e listar os elementos resultantes do projeto de teste.

### **1.2 O Cronos**

O sistema CRONOS consiste em uma ferramenta que receba como entrada uma base de dados de qualquer domínio em um formato pré-definido (CSV) e irá gerar arquivos com bases menores em determinado intervalo de tempo. A principal característica que esse Dataset deve possuir é a existência de pelo menos um atributo que seja temporal.

Portanto, a missão do software consiste em possibilitar a criação de janelas temporais móveis, com intervalos definidos dinamicamente pelo usuário, afim de facilitar o processo de particionamento de datasets.

### **1.3 Escopo**

O sistema CRONOS passará pelos testes de unidade, de integração e de sistema. Os testes de unidade consistirão no teste de maneira isolada, geralmente simulando as dependências que cada unidade tem. Já os testes de integração do sistema são aqueles que testam a integração entre duas partes do sistema. Para a execução dos testes serão utilizados arquivos temporais com formato csv para manipulação dos dados.

## **2. Requisitos a Testar**

A lista abaixo identifica aqueles itens – use cases, requisitos funcionais e não funcionais que foram identificados como alvos de teste. Essa lista representa o que será testado.

### **2.1 Teste do Integridade de Dados**

- Verificar se os dados contidos sub-bases geradas pelo sistema estão de acordo com a base original
- Verificar se o sistema é capaz de salvar os dados nos diretórios corretos
- Verificar se o sistema é capaz de particionar a base de dados seguindo o intervalo de tempo especificado
- Verificar se o sistema é capaz de nomear as bases geradas conforme o que foi especificado pelo usuário.

### **2.2 Teste Funcional**

- Verificar se o usuário pode carregar seu arquivo de tipo CSV
- Verifique que o sistema é capaz de ler os atributos
- Verifique que o sistema possibilita a escolha do formato do atributo temporal que será utilizado na base de dados

### **2.3 Teste da Interface do Usuário**

- Navegue através de todos os casos de uso, verificando que cada tela de interface gráfica pode ser rapidamente entendida e facilmente utilizada.
- Verifique que todas as palavras e expressões apresentadas estão em conformidade com as devidas normas sintáticas e gramaticais.

### **2.4 Perfil da Performance**

- Verifique que o tempo de resposta do sistema quando solicitado o particionamento não ultrapassa 1 Minuto (com um arquivo de tamanho médio).

### **2.5 Teste de Instalação**

- Verifique que a instalação do aplicativo foi bem sucedida.

### 3. Estratégia de Teste

Nesta seção serão descritos os tipos de teste a serem aplicados no sistema, incluindo critérios de parada, técnicas utilizadas e considerações especiais para o teste da funcionalidade, quando houver.

#### 3.1 Tipos de Teste

Nota: As transações abaixo se referem às “transações lógicas de negócio”. Essas transações são definidas como funções específicas que um usuário final do sistema é suposto de executar ao usar a aplicação, tais como carregar um arquivo ou selecionar seu diretório de salvamento.

##### 3.1.1 Teste de Integridade de Dados

Objetivo do Teste:	Garantir que os registros salvos nas sub-bases coincidem com os dados originários da base de dados que está sendo dividida.
Técnica:	<ul style="list-style-type: none"><li>▪ Inspecionar os arquivos de saída para garantir que os dados foram populados como pretendido, que todos os eventos de salvamento ocorreram apropriadamente, ou revisar os dados retornados para garantir que os dados corretos gravados no diretório correto, no intervalo de tempo especificado e com o nome solicitado pelo usuário.</li></ul>
Critério de Finalização:	<ul style="list-style-type: none"><li>▪ Todos os métodos e processos de acesso aos arquivos .csv funcionam como projetados e sem nenhuma corrupção de dados.</li></ul>
Considerações Especiais:	<ul style="list-style-type: none"><li>▪ Processos devem ser invocados manualmente.</li></ul>

### 3.1.2 Teste de Função

Objetivo do Teste:	Garantir a funcionalidade apropriada do alvo do teste, incluindo navegação, entrada de dados, processamento, e recuperação.
Técnica:	<ul style="list-style-type: none"><li>▪ Executar cada caso de uso, fluxo de caso de uso, usando dados válidos e inválidos, para verificar o seguinte:</li><li>▪ Os resultados esperados ocorrem quando dados válidos são usados</li><li>▪ As mensagens de erro ou aviso apropriadas são exibidas quando dados inválidos são usados.</li></ul>
Critério de Finalização:	<ul style="list-style-type: none"><li>▪ Todos os testes planejados foram executados.</li><li>▪ Todos os defeitos identificados foram tratados.</li></ul>
Considerações Especiais:	Nenhum

### 3.1.3 Teste da Interface do Usuário

Objetivo do Teste:	Identificar se: <ul style="list-style-type: none"><li>▪ A navegação através dos alvos de teste reflete as funções e os requisitos do negócio apropriadamente, incluindo janela-a-janela, campo-a-campo.</li><li>▪ Objetos e características da janela, tais como menus, tamanho, posição, estado e foco conformam-se aos padrões.</li></ul>
Técnica:	<ul style="list-style-type: none"><li>▪ Criar ou modificar os testes para cada janela para verificar a navegação e os estados de objeto apropriados para cada janela e objetos da aplicação.</li></ul>

Critério de Finalização:	<ul style="list-style-type: none"> <li>▪ É verificado que cada janela permanece consistente com a versão de comparação ou dentro de padrões aceitáveis.</li> </ul>
Considerações Especiais:	<ul style="list-style-type: none"> <li>▪ Nem todas as propriedades para objetos personalizados e terceirizados podem ser acessadas.</li> </ul>

### 3.1.4 Teste de Performance

Objetivo do Teste:	Verificar o tempo de resposta.
Técnica:	<ul style="list-style-type: none"> <li>▪ Cronometrar o tempo de resposta, quando ocorrer uma transação (Particionamento).</li> </ul>
Critério de Finalização:	<ul style="list-style-type: none"> <li>▪ Todas as transações de teste foram efetuadas e o seu tempo de resposta foi cronometrado.</li> </ul>
Considerações Especiais:	<ul style="list-style-type: none"> <li>▪ O teste de performance requer um cronômetro.</li> <li>▪ Considerar também o número de campos e de registros da base de dados particionada.</li> </ul>



### 3.1.5 Teste de Instalação

Objetivo do Teste:	Verifique que os alvos de teste instalam apropriadamente em cada configuração de hardware necessária sobre as seguintes condições: <ul style="list-style-type: none"><li>▪ Windows 7 ou superior – Linux (All Distributions).</li><li>▪ JRE 1.6 - Superior</li></ul>
Técnica:	Começar ou executar a instalação em diferentes tipos de dispositivos de diferentes versões do sistema operacional.
Critério de Finalização:	As transações do APP executam de forma bem sucedida, sem falha em dispositivos que atendem a requisição mínima, e falha naqueles que não possuem.
Considerações Especiais:	Nenhuma.

## 4. Recursos

Essa seção apresenta os recursos recomendados para o projeto do sistema Cronos, suas principais responsabilidades, e seus conhecimentos.

## 4.1 Trabalhadores

Essa tabela mostra as suposições de recrutamento para o projeto.

Recursos Humanos		
Trabalhador	Recursos Mínimos Recomendados	Responsabilidades Específicas ou Comentários
Cleyciane Farias Juliana Abreu Raul Vitor L		Define e descreve os cenários de casos de testes.  Responsabilidades: <ul style="list-style-type: none"><li>▪ Gerar o plano de teste</li><li>▪ Criar o plano de teste</li><li>▪ Avaliar o teste executado</li></ul>
Cleyciane Farias Juliana Abreu Raul Vitor L		Executa os testes criados.  Responsabilidades: <ul style="list-style-type: none"><li>▪ Executar os testes que foram desritos</li><li>▪ Registrar os resultados obtidos com o teste</li><li>▪ Observar erros que foram percebidos durante a obtenção dos resultados</li><li>▪ Solicita, Descreve as mudanças que foram executadas no plano de teste</li></ul>
Cleyciane Farias Juliana Abreu Raul Vitor L		Implementa e faz os testes unitários das classes e pacotes de teste.  Responsabilidades: <ul style="list-style-type: none"><li>▪ Criar as classes e pacotes de teste implementados no modelo de teste para o sistema CRONOS</li></ul>

## 4.2 Sistema

A tabela seguinte expõe os recursos do sistema para o projeto de teste.

Recursos do Sistema
Interface Gráfica Java: Swing Java: AWT
Dispositivos 3 PCs: Windows 7, Windows 10-Ubuntu, Linux Mint.
Repositório de Testes Django.csv Teste.csv

## 5. Cronograma

Atividade	Data de Início	Data de Término
Teste de Instalação.	12/06/2019	12/06/2019
Teste de Função, Teste de Integridade de Dados.	12/06/2019	12/06/2019
Teste de Performance, Teste da Interface do Usuário	12/06/2019	12/06/2019