

Centro Universitário União das Américas Descomplica Engenharia de
Software IV

Plano de Teste

Guerin – Sistema de Controle e Gestão de Gado

1.Introdução

1.1 Propósito do Sistema

A fazenda Guerin é uma fazenda de criação de gado. O demandante necessita de um sistema web que proporcione maior facilidade e eficiência no controle e gestão do gado.

A solução será desenvolver um sistema onde seja possível realizar todas as tarefas relacionadas a gestão do gado.

O sistema permitirá os gerenciamentos dos usuários e gados, como cadastrar, alterar, pesquisar e excluir. Possibilitará uma listagem dos gados, vacinas e atividades relacionadas ao gado e realizar os filtros necessários.

1.2 Referências/Documentações do Projeto

A tabela abaixo identifica a documentação utilizada para a elaboração deste Plano de Teste:

Documento	Irá fazer parte do projeto? (Sim ou Não)	Observações
Diagrama de Classes	Sim	Irá definir as classes que o sistema deve possuir.
Especificação de Casos de Uso	Sim	Descreverá de forma detalhada as funcionalidades do sistema.
Especificação das Regras de Negócio	Sim	Definirá as regras e os comportamentos do sistema.

2.Abordagem de Testes

2.1 Categorização dos Requisitos em Funcionais x Não Funcionais

Requisito Funcional	Requisito Não Funcional
RF001 – O sistema deverá permitir o cadastro de usuários.	NF001 – O sistema deverá ser desenvolvido na linguagem de programação Java.

RF002 – O sistema deve permitir a alteração de informações referentes ao usuário cadastrado.	NF002 – O banco de dados utilizado deverá ser o PostgreSQL.
RF003 – O sistema deve permitir a busca de um usuário cadastrado.	NF003 – O sistema deverá ser compatível com o sistema operacional Windows.
RF004 – O sistema deve permitir a exclusão de um usuário cadastrado.	NF004 - O sistema deve ser acessível via Browser, como Google Chrome, Internet Explorer e Mozilla Firefox.
RF005 – O sistema deve permitir o cadastro de gados.	
RF006 – O sistema deve permitir a alteração de informações referentes ao gado cadastrado.	
RF007 – O sistema deve permitir a busca de um gado cadastrado.	
RF008 – O sistema deve permitir a inativação de um gado cadastrado.	
RF009 – O sistema deve permitir o cadastro de vacinas.	
RF010 – O sistema deve permitir a alteração de informações referentes à vacina cadastrada.	
RF011 – O sistema deve permitir a busca de uma vacina cadastrada.	
RF012 – O sistema deve permitir a inativação de uma vacina cadastrada.	
RF013 – O sistema deve permitir o cadastro de um evento relacionado ao gado.	
RF014 – O sistema deve permitir a alteração de informações de um evento relacionado ao gado.	
RF015 – O sistema deve permitir a busca de um evento relacionado ao gado cadastrado.	
RF016 – O sistema deve permitir a inativação de um evento relacionado ao gado cadastrado.	
RF017 – O sistema deve permitir o cadastro de espécie de gado.	
RF018 – O sistema deve permitir a alteração de informações de uma espécie cadastrada.	
RF019 – O sistema deve permitir a busca de uma espécie cadastrada.	
RF020 – O sistema deve permitir a inativação de uma espécie cadastrada.	
RF021 – O sistema deve possuir filtros dos brincos do pai e mãe dos gados.	
RF022 – O sistema deve possuir níveis de acesso para diferenciar usuários comuns de	

usuários administrativos, que acessarão áreas restritas do sistema.	
RF023 – O sistema deve permitir o cadastro de menu	
RF024 – O sistema deve permitir a busca dos menus cadastrados.	
RF025 – O sistema deve possuir a busca de um menu cadastrado a partir do nome do mesmo	
RF026 – O sistema deve possuir a busca de um menu cadastrado a partir do Id do mesmo	
RF027 – O sistema deve possuir a função de desativar um menu a partir de seu ID	

2.2 Detalhamento da Abordagem de Teste

Tipo de Teste:	Funcional
Subtipo de Teste:	Requisitos
Objetivo do Teste:	Testar as funcionalidades do controle do usuário
Funcionalidades de Teste	Inserir, Editar, Buscar e Desativar

Requisitos que motivaram esse teste:	<p>RF001 – O sistema deverá permitir o cadastro de usuários.</p> <p>RF002 – O sistema deve permitir a alteração de informações referentes ao usuário cadastrado.</p> <p>RF003 – O sistema deve permitir a busca de um usuário cadastrado.</p> <p>RF004 – O sistema deve permitir a exclusão de um usuário cadastrado.</p>
---	---

Tipo de Teste:	Funcional
Subtipo de Teste:	Requisitos
Objetivo do Teste:	Testar as funcionalidades do controle do Gado
Funcionalidades de Teste	Inserir, Editar, Buscar e Desativar
Requisitos que motivaram esse teste:	<p>RF005 – O sistema deve permitir o cadastro de gados.</p> <p>RF006 – O sistema deve permitir a alteração de informações referentes ao gado cadastrado.</p> <p>RF007 – O sistema deve permitir a busca de um gado cadastrado.</p> <p>RF008 – O sistema deve permitir a inativação de um gado cadastrado.</p>

Tipo de Teste:	Funcional
Subtipo de Teste:	Requisitos

Objetivo do Teste:	Testar as funcionalidades do controle das vacinas
Funcionalidades de Teste	Inserir, Editar, Buscar e Desativar
Requisitos que motivaram esse teste:	<p>RF009 – O sistema deve permitir o cadastro de vacinas.</p> <p>RF010 – O sistema deve permitir a alteração de informações referentes à vacina cadastrada.</p> <p>RF011 – O sistema deve permitir a busca de uma vacina cadastrada.</p> <p>RF012 – O sistema deve permitir a inativação de uma vacina cadastrada.</p>

Tipo de Teste:	Funcional
Subtipo de Teste:	Requisitos
Objetivo do Teste:	Testar as funcionalidades do controle dos eventos
Funcionalidades de Teste	Inserir, Editar, Buscar e Desativar
Requisitos que motivaram esse teste:	<p>RF013 – O sistema deve permitir o cadastro de um evento relacionado ao gado.</p> <p>RF014 – O sistema deve permitir a alteração de informações de um evento relacionado ao gado</p> <p>RF015 – O sistema deve permitir a busca de um evento relacionado ao gado cadastrado.</p> <p>RF016 – O sistema deve permitir a inativação de um evento relacionado ao gado cadastrado.</p>

Tipo de Teste:	Funcional
Subtipo de Teste:	Requisitos
Objetivo do Teste:	Testar as funcionalidades do controle das vacinas
Funcionalidades de Teste	Inserir, Editar, Buscar e Desativar
Requisitos que motivaram esse teste:	<p>RF017 – O sistema deve permitir o cadastro de espécie de gado.</p> <p>RF018 – O sistema deve permitir a alteração de informações de uma espécie cadastrada.</p> <p>RF019 – O sistema deve permitir a busca de uma espécie cadastrada.</p> <p>RF020 – O sistema deve permitir a inativação de uma espécie cadastrada.</p>

Tipo de Teste:	Funcional
Subtipo de Teste:	Requisitos
Objetivo do Teste:	Testar as funcionalidades do controle de alguns filtros dentro do sistema.
Funcionalidades de Teste	Filtros para controle de gado
Requisitos que motivaram esse teste:	RF021 – O sistema deve possuir filtros dos brincos do pai e mãe dos gados.

Tipo de Teste:	Funcional
Subtipo de Teste:	Segurança
Objetivo do Teste:	Testar as funcionalidades do controle de alguns filtros dentro do sistema.
Funcionalidades de Teste	Filtros para controle de gado
Requisitos que motivaram esse teste:	RF022 – O sistema deve possuir níveis de acesso para diferenciar usuários comuns de usuários administrativos, que acessarão áreas restritas do sistema.

Tipo de Teste:	Funcional
Subtipo de Teste:	Segurança
Objetivo do Teste:	Testar as funcionalidades do controle de alguns filtros dentro do sistema.
Funcionalidades de Teste	Filtros para controle de gado
Requisitos que motivaram esse teste:	<p>RF023 – O sistema deve permitir o cadastro de menu.</p> <p>R024 - O sistema deve permitir a busca dos menus cadastrados</p> <p>R025 - O sistema deve possuir a busca de um menu cadastrado a partir do nome do mesmo</p>

	<p>RF026 - O sistema deve possuir a busca de um menu cadastrado a partir do Id do mesmo</p> <p>RF027 - O sistema deve possuir a função de desativar um menu a partir de seu ID</p>
--	--

Tipo de Teste:	Não Funcional
Subtipo de Teste:	Linguagem de Desenvolvimento
Objetivo do Teste:	Verificar se o sistema está sendo codificado na linguagem java.
Requisitos que motivaram esse teste:	NF001 – O sistema deverá ser desenvolvido na linguagem de programação Java.

Tipo de Teste:	Não Funcional
Subtipo de Teste:	Sistema de Dados
Objetivo do Teste:	Testar o banco de dados que deverá ser usado no desenvolvimento do sistema.
Requisitos que motivaram esse teste:	NF002 – O banco de dados utilizado deverá ser o PostgreSQL.

Tipo de Teste:	Não Funcional
-----------------------	---------------

Subtipo de Teste:	Sistema Operacional Compatível
Objetivo do Teste:	Criação de um sistema que será principalmente compatível com o sistema operacional Windows.
Requisitos que motivaram esse teste:	NF003 – O sistema deverá ser compatível com o sistema operacional Windows.

Tipo de Teste:	Não Funcional
Subtipo de Teste:	Acesso do Sistema
Objetivo do Teste:	Criação de um sistema que será utilizado nos navegadores mais comuns e de acesso a internet.
Requisitos que motivaram esse teste:	NF004 - O sistema deve ser acessível via Browser, como Google Chrome, Internet Explorer e Mozilla Firefox.

2.3 Ferramentas

As seguintes ferramentas serão empregadas neste projeto de testes:

Ferramenta	Fabricante	Versão
IntelliJ	JetBrains	2021.3.3
JUnit Jupiter API	JUnit.org	5.8.2
Mockito Framework	Mockito.org	4.7.0

3.Ambiente de Teste

3.1 Definições do Ambiente de Teste

- Dentro do ambiente de desenvolvimento, os testes unitários inicialmente são feitos pelos desenvolvedores de forma simples, ou seja, utilizando uma grade de pequenos dados e fazendo assim seus testes mais focados em partes.
- Após feito os testes mais focalizados, se é iniciado os testes mais gerais em toda aplicação com uma quantidade maior de dados reais, para poder ver como que o programa irá se comportar.
- Os testes deverão ser feitos na máquina dos desenvolvedores conectados via conexão de internet.
- O sistema na parte do backend será desenvolvido na linguagem Java (SpringBoot) com auxílio do postgresSQL.

4.Programação dos Testes

4.1 Objetivos e Prioridades

As prioridades do sistema são:

- O sistema possibilitará ao usuário o gerenciamento dos gados do demandante;
- O sistema possibilitará ao usuário detalhar todas as informações dos gados presente em sua fazenda;
- O sistema deve possuir um design web simples e responsivo;
- O sistema possibilitará o controle de eventos relacionados aos gados;
- O sistema possibilitará o controle de vacinas aplicadas ao gado
- O sistema possibilitará o demandante ter diferentes fazendas com a gestão dos gados referentes a mesma;

5.Casos de Teste

Casos	Teste	Passos	Resultado Esperado
Cattle	Entity	Verificar se o atributo inactive tem o valor "false" por padrão	Atributo inactive iniciar com o valor "false"
		Verificar se o brinco do pai do gado está correto	Brinco do pai do gado tenha o mesmo valor que foi inserido no construtor

		Verificar se o brinco da mãe do gado está correto	Brinco da mãe do gado tenha o mesmo valor que foi inserido no construtor
		Verificar se o peso do gado está correto	Peso do gado tem o mesmo valor que foi inserido no construtor
		Verificar se a Espécie do gado está correto	Especie do gado tem o mesmo valor que foi inserido no construtor
		Verificar se a Fazenda do gado está correto	Fazenda do gado tem o mesmo valor que foi inserido no construtor
		Verificar se o gênero do gado está correto	Gênero do gado tem o mesmo valor que foi inserido no construtor
CattleEvent	Entity	Verificar se o valor do atributo inactive é falso	O valor do atributo ser false.
		Verificar se há erro na entrada de dados do atributo gado.	O valor do atributo não ser nulo
		Verificar se há erro na entrada de dados do atributo tipo do evento.	O valor do atributo não ser nulo.

		Verificar se a data do evento é maior que a atual (data futura)	A data do evento ser menor que a atual
		Verificar se a entrada de dados da aplicação de vacina está correto	O atributo aplicação de vacina não ser nulo.
		Verificar se a entrada de dados do atributo pesagem está correto	O atributo pesagem não ser nulo
EventType	Entity	Verificação de inativo	Verifica se o EventType em questão esta ativo ou inativo;
		Verificação de Igualdade	Verifica se dois objetos do tipo EventType são iguais.
Farm	Entity	Verificar se o atributo inactive tem o valor “false” por padrão	Atributo inactive iniciar com o valor “false”
		Verificar se o nome da fazenda está correto	Nome da fazenda tem o mesmo valor que foi inserido no construtor
		Verificar se o endereço da fazenda está correto	Endereço da fazenda tem o mesmo valor que foi inserido no construtor
Specie	Entity	Inactive Teste	Verificar se a Specie está inativa;

		Comparação de dois Objetos Specie	Verificar se os valores dos dois objetos são iguais
User	Entity	Inactive Teste	Verificar se o valor Inactive é false
		Comparação de Valores	Verificar se os valores inseridos foram corretos
VaccineApplication	Entity	Verificar se o atributo inactive não é false por default	Retornar false ao comparar o atributo
		Verificar se a data da aplicação é maior que a atual (data futura)	A data ser menor do que a data atual
		Verificar se a entrada de dados do atributo gado está correta	O atributo gado não ser nulo
		Verificar se a entrada de dados do atributo vacina da aplicação está correto	A vacina não ser nulo
Vaccine	Entity	Verificar se o atributo inactive não é false por default	Retornar false ao comparar o atributo
		Verificar se a data é maior que a atual (data futura)	A data ser menor do que a data atual
Weighing	Entity	Inactive Teste	Verificar se o valor Inactive é false

		Comparação de Valores	Verificar se os valores inseridos foram corretos
		Valores Preenchidos	Verificar se os valores foram inseridos de forma corretamente, não sendo nulos
Cattle	Service	Verificar se o gado foi inserido no banco de dados	Um objeto Cattle igual ao que foi cadastrado no banco de dados
		Verificar se o gado foi atualizado no banco de dados	Um objeto Cattle com os mesmos atributos que foram atualizado no banco de dados
		Verificar se os pais do gado são os mesmos que estão sendo relacionados com o próprio gado.	Objetos Cattle pai e Cattle mãe iguais aos que estão sendo relacionados com o Cattle filho
		Verificar se os filhos do gado são os mesmos que estão sendo relacionados com o próprio gado (pai ou mãe)	Lista de objetos Cattle que tem relação com com o gado (pai ou mãe) informado
		Verificar se o gado requisitado pelo id é igual ao cadastrado no banco de dados	Um objeto Cattle igual ao que foi requisitado

		Verificar se o gado requisitado pelo brinco é igual ao cadastrado no banco de dados	Um objeto Cattle igual ao que foi requisitado
		Verificar se o gado requisitado pelo brinco é igual ao cadastrado no banco de dados	Um objeto Cattle igual ao que foi requisitado ou uma nova instância em caso de não encontrar o objeto.
		Verificar se o gado requisitado pela espécie é igual ao cadastrado no banco de dados	Um objeto Cattle igual ao que foi requisitado
		Verificar se o gado requisitado pela fazenda é igual ao cadastrado no banco de dados	Um objeto Cattle igual ao que foi requisitado
		Verificar se todos os gados cadastrados foram encontrados	Um número de objetos retornados igual ao de gados cadastrados
		Verificar se o gado está inativo	Um objeto Cattle com o atributo Inactive com o valor “true”

CattleEvent	Service	Cadastrar um evento com o tipo vacinação e verificar o cadastramento do mesmo no banco de dados através de uma busca.	Retornar o mesmo objeto cadastrado
		Cadastrar um evento com o tipo pesagem e verificar o cadastramento do mesmo no banco de dados através de uma busca.	Retornar o mesmo objeto cadastrado
		Cadastrar um evento com um tipo aleatório e verificar o cadastramento do mesmo no banco de dados através de uma busca.	Retornar o mesmo objeto cadastrado
		Fazer a listagem dos eventos do gado	Retornar a lista com todos os eventos cadastrados
		Fazer um cadastramento de um evento, atualizar o mesmo e depois fazer uma busca para verificação	Retornar o objeto atualizado

		Fazer o cadastramento de um evento, desabilitá-lo e fazer uma busca do mesmo para verificação do status de atividade	Retornar o objeto com o status inactive igual a true
		Fazer o cadastramento de um evento e realizar uma busca do tipo do evento que foi cadastrado no evento	Retornar o tipo de evento cadastrado no objeto
		Fazer o cadastramento de um evento e realizar uma busca da paisagem que foi cadastrada no evento	Retornar a paisagem que foi cadastrada no evento
		Fazer o cadastramento de um evento e realizar uma busca da aplicação de vacina que foi cadastrada no evento	Retornar a aplicação de vacina que foi cadastrada no evento
		Fazer o cadastramento de um evento e realizar uma busca do gado que foi cadastrado no evento	Retornar o gado que foi cadastrado no evento

EventType	Service	Cadastrar um EventType	Deve obter o retorno de que o EventType em questão foi registrado no banco;
		Atualizar um EventType	Deve se obter o retorno de que a propriedade do EventType foi alterada no banco de dados;
		Desativar um EventType	Deve se obter o retorno de que a propriedade inativa do EventType foi alterada para true no banco de dados;
		Encontrar um EventType pelo ID	Deve se obter o retorno do Objeto EventType que representa esse ID no banco de dados
		Encontrar todos os EventType cadastrados no banco	Essa busca deve retornar corretamente todos os EventType que estão registrados no banco de dados;
Farm	Service	Verificar se a fazenda foi inserida no banco de dados	Um objeto Farm igual ao que foi cadastrado no banco de dados
		Verificar se a fazenda foi atualizada no banco de dados	Um objeto Farm com os mesmos atributos que foram atualizado no banco de dados

		Verificar se a fazenda requisitada pelo nome é igual a cadastrada no banco de dados	Um objeto Farm igual ao que foi requisitado
		Verificar se a fazenda requisitada pelo endereço é igual a cadastrada no banco de dados	Um objeto Farm igual ao que foi requisitado
		Verificar se a fazenda requisitada pelo id é igual a cadastrada no banco de dados	Um objeto Farm igual ao que foi requisitado
		Verificar se todas as fazendas cadastradas foram encontradas	Um número de objetos retornados igual ao de fazendas cadastradas
		Verificar se a fazenda está inativa	Um objeto Farm com o atributo Inactive com o valor “true”
Specie	Service	Cadastrar uma Specie	Deve se obter o retorno de que o Objeto Specie foi cadastrado no banco de dados.

		Atualizar uma Specie	Deve se obter o retorno de que a Specie em questão tenha seus dados alterados no banco de dados.
		Desativar uma Specie	Deve se obter o retorno de que a Specie em questão foi desativada dentro do Banco de Dados;
		Encontrar Specie por ID	Deve se obter o retorno da Specie que possui o ID inserido;
		Encontrar Todas as Specie Cadastradas	Deve se obter o retorno de todas as Specie do banco de dados;
User	Service	Insert	O inserimento dos valores de forma correta no banco de dados
		Update	A edição dos valores de forma correta no banco de dados
		FindAll	Busca de todos os valores no banco de dados

		FindByUsername	Busca de valores pelo Username no banco de dados
		FindById	Busca de valores pelo ID no banco de dados
		FindByEmail	Busca de valores pelo Email no banco de dados
		LoadUserByUsername	Um princípio do teste de login, se os parametros estão corretos para efetuar o acesso
		Disable	A desabilitação dos valores no banco de dados, focando no inactive, assim não deletando totalmente do banco de dados
VaccineApplication	Service	Ao salvar uma aplicação de vacina, verificar o cadastramento da mesma no banco de dados ao fazer um busca	Retorno do mesmo objeto cadastrado

		Verificar a listagem de todas as aplicações de vacinas	Retorno da lista de aplicações de vacina
		Ao cadastrar uma aplicação de vacina, atualizar um atributo dela e verificar se foi atualizado no banco de dados	Retorno do objeto com os dados atualizados
		Ao cadastrar uma aplicação de vacina, desabilitar o mesmo e verificar a mudança de status no banco de dados	Retorno do objeto com o status inactive igual a true
Vaccine	Service	Ao salvar uma vacina, verificar o cadastramento da mesma no banco de dados ao fazer um busca	Retorno do mesmo objeto cadastrado
		Verificar a listagem de todas as vacinas	Retorno da lista vacinas
		Ao cadastrar uma vacina, atualizar um atributo dela e verificar se foi atualizado no banco de dados	Retorno do objeto com os dados atualizados
		Ao cadastrar uma vacina, desabilitar o mesmo e verificar a	Retorno do objeto com o status inactive igual a true

		mudança de status no banco de dados	
Weighing	Service	Insert	O inserimento dos valores de forma correta no banco de dados
		Update	A edição dos valores de forma correta no banco de dados
		FindAll	Busca de todos os valores no banco de dados
		FindById	Busca de valores pelo ID no banco de dados
		Disable	A desabilitação dos valores no banco de dados, focando no inactive, assim não deletando totalmente do banco de dados
CattleEvent	Controller	Realizar a busca de um evento pelo id	Retornar o evento correspondente ao id da busca
		Realizar o desabilitamento de um evento passando o id do mesmo como parametro	Retorno do objeto com o status inactive igual a true

		Realizar o cadastramento de um evento	Retorno do mesmo objeto cadastrado
		Realizar a listagem dos eventos cadastrados	Retorno de uma lista com os eventos cadastrados
		Realizar a busca de um eventos passando como parametro o id do tipo de evento	Retorno de uma lista dos eventos que possuem aquele mesmo tipo de evento
		Realizar a busca de eventos passando como parametro o id da pesagem	Retorno de uma lista de eventos que possuem pesagem
		Realizar a busca de uma aplicação de vacina passando como parametro o id da aplicação de vacina	Retorno da aplicação de vacina correspondente ao id passado
		Realizar a busca dos eventos passando como parametro o id de um determinado gado	Retorno de uma lista de eventos vinculados ao gado
VaccinationApplication	Controller	Realizar a busca de uma aplicação de vacina pelo id	Retornar a aplicação de vacina correspondente ao id da busca
		Realizar o desabilitamento de uma aplicação de vacina passando o id da mesma como parametro	Retorno do objeto com o status inactive igual a true
		Realizar o cadastramento de uma aplicação de vacina	Retorno do mesmo objeto cadastrado
		Realizar a listagem das aplicações de vacinas cadastradas	Retorno de uma lista com as aplicações de vacina cadastradas

		Realizar uma busca das aplicações de vacina passando como parametro o id de uma determinada vacina	Retorno de uma lista de aplicações cadastradas com o id da vacina buscada
		Realizar uma atualização de uma aplicação de vacina cadastrada e verificar se foi atualizada	Retorno da aplicação de vacina atualizada
Vaccine	Controller	Realizar a busca de uma vacina pelo id	Retornar a vacina correspondente ao id da busca
		Realizar o desabilitamento de uma vacina passando o id da mesma como parametro	Retorno do objeto com o status inactive igual a true
		Realizar a listagem das vacinas cadastradas	Retorno de uma lista com as vacinas cadastradas
		Realizar o cadastramento de uma vacina	Retorno do mesmo objeto cadastrado
		Realizar a busca de uma vacina passando como parametro um nome	Retorno da vacina que possui o nome passado como parametro
Weighing	Controller	Listar todos os dados dentro do banco de dados	Retornar todos os dados que estão inseridos no banco de dados
		Listar apenas o ID desejado dentro do banco de dados	Retornar apenas a pesagem correspondente ao ID desejado

		Inserir a pegagem dentro do banco de dados	A inserção do objeto dentro do banco de dados
		Buscar e Desativar a pesagem existente escolhida a partir do ID	Retornar o desativamento do objeto, que foi passado o ID como parametro
Cattle	Controller	Verificar se o gado foi inserido no banco de dados	Um objeto Cattle igual ao que foi cadastrado no banco de dados
		Verificar se o gado foi atualizado no banco de dados	Um objeto Cattle com os mesmos atributos que foram atualizado no banco de dados
		Verificar se os pais do gado são os mesmos que estão sendo relacionados com o próprio gado.	Objetos Cattle pai e Cattle mãe iguais aos que estão sendo relacionados com o Cattle filho
		Verificar se os filhos do gado são os mesmos que estão sendo relacionados com o próprio gado (pai ou mãe)	Lista de objetos Cattle que tem relação com com o gado (pai ou mãe) informado

		Verificar se o gado requisitado pelo id é igual ao cadastrado no banco de dados	Um objeto Cattle igual ao que foi requisitado
		Verificar se o gado requisitado pelo brinco é igual ao cadastrado no banco de dados	Um objeto Cattle igual ao que foi requisitado
		Verificar se o gado requisitado pelo brinco é igual ao cadastrado no banco de dados	Um objeto Cattle igual ao que foi requisitado ou uma nova instância em caso de não encontrar o objeto.
		Verificar se o gado requisitado pela espécie é igual ao cadastrado no banco de dados	Um objeto Cattle igual ao que foi requisitado
		Verificar se o gado requisitado pela fazenda é igual ao cadastrado no banco de dados	Um objeto Cattle igual ao que foi requisitado
		Verificar se todos os gados cadastrados foram encontrados	Um número de objetos retornados igual ao de gados cadastrados

		Verificar se o gado está inativo	Um objeto Cattle com o atributo Inactive com o valor “true”
Farm	Controller	Verificar se a fazenda foi inserida no banco de dados	Um objeto Farm igual ao que foi cadastrado no banco de dados
		Verificar se a fazenda foi atualizada no banco de dados	Um objeto Farm com os mesmos atributos que foram atualizado no banco de dados
		Verificar se a fazenda requisitada pelo nome é igual a cadastrada no banco de dados	Um objeto Farm igual ao que foi requisitado
		Verificar se a fazenda requisitada pelo endereço é igual a cadastrada no banco de dados	Um objeto Farm igual ao que foi requisitado
		Verificar se a fazenda requisitada pelo id é igual a cadastrada no banco de dados	Um objeto Farm igual ao que foi requisitado

		Verificar se todos as fazendas cadastradas foram encontradas	Um número de objetos retornados igual ao de fazendas cadastradas
		Verificar se a fazenda está inativa	Um objeto Farm com o atributo Inactive com o valor “true”
User	Controller	Verificar se o usuario foi inserido no banco de dados	Um objeto User igual ao que foi cadastrado no banco de dados
		Verificar se o usuario foi atualizado no banco de dados	Um objeto User com os mesmos atributos que foram atualizado no banco de dados
		Verificar se o usuario consultado pelo username é igual ao cadastrado no banco de dados	Um objeto User igual ao que foi requisitado
		Verificar se o usuario consultado pelo email é igual ao cadastrado no banco de dados	Um objeto User igual ao que foi requisitado

		Verificar se o usuario requisitado pelo id é igual ao cadastrada no banco de dados	Um objeto User igual ao que foi requisitado
		Verificar se todos os usuarios cadastrados foram encontrados	Um número de objetos retornados igual ao de usuarios cadastrados
		Verificar se o usuario está inativo	Um objeto User com o atributo Inactive com o valor “true”
		Verificar se o refresh token está vindo corretamente	Um novo access_token valido, utilizando um refresh token valido, ou uma mensagem de erro caso for um refresh_token invalido ou vencido
Parameter	Controller	Verificar se o parametro foi inserido no banco de dados	Um objeto Parameter igual ao que foi cadastrado no banco de dados

		Verificar se o parametro foi atualizado no banco de dados	Um objeto Parameter com os mesmos atributos que foram atualizado no banco de dados
		Verificar se o parametro requisitado pelo id é igual ao cadastrado no banco de dados	Um objeto Parametro igual ao que foi requisitado
		Verificar se todos os parametros cadastrados foram encontrados	Um número de objetos retornados igual ao de parametros cadastrados
		Verificar se o parametro foi excluido com sucesso	Retornar null na requisição por id, depois de realizar a exclusão
MENU	SERVICE	Verificar se o menu foi inserido no banco	Um objeto menu igual ao que foi inserido no banco de dados

		Verificar o Id do Registro retorna o menu registrado no banco	Um objeto igual ao requisitado
		Verificar se o Nome do Registro retorna o menu registrado no banco	Um objeto igual com o nome igual ao requisitado
		Verificar se o Objeto foi desativado no banco	Verificar se o objeto requisitado por Id está com seu atributo inativo como true
		Verificar há possibilidade de duplicação de menu	Verificar se o menu está duplicado dentro do banco de dados
MENU	CONTROLLER	Verificar se os parametros foram inserido no banco a partir da rota	Mandar uma requisição post e verificar se os dados foram inseridos de acordo.

		Verificar o retorno de todos os menu registrados no banco	Verificar se o get da rota retorna todos os menus registrados no banco de dados
		Verificar o retorno de um menu por Id	Verificar se o get da rota com o Id retorna o o objeto desejado
		Verificar o retorno de um menu por Nome	Verificar se o get da rota com o Nome do menu retorna o objeto desejado
		Verificar a desativação do menu por rota.	Verificar se o objeto desejado tem seu atributo de inactive alterado para true