

Tecnologia em Análise e Desenvolvimento de Sistemas

Gabriel Giovani Felicidade

Giovanni Garcia Hessel

Guilherme Alves Martins de Aguiar

Guilherme Igor Alves Sernajoto

Documentação de Desenvolvimento de Software

Sistema de Gestão de Patrimônio

Sorocaba

Julho - 2019

Gabriel Giovani Felicidade

Giovanni Garcia Hessel

Guilherme Alves Martins de Aguiar

Guilherme Igor Alves Sernajoto

Documentação de Desenvolvimento de Software

Sistema de Gestão de Patrimônio

Trabalho de Graduação apresentado à Faculdade de Tecnologia de Sorocaba – FATEC, como parte dos pré-requisitos para obtenção do título de Tecnólogo em Análise e Desenvolvimento de Sistemas

Orientador: Prof. Renato Jensen

Sorocaba

Julho – 2019

**Dedicatória**

Exemplo de dedicatória

Dedico este trabalho aos meus pais que com muita perseverança e paciência sempre incentivaram meus estudos e proporcionaram a educação que hoje tenho.

**Agradecimento**

Exemplo de agradecimento

Agradeço a todos que de uma forma ou de outra colaboraram para com este trabalho,

em especial aos meus grandes amigos Luke Skywalker e seu pai Darth Vader que me ajudaram com a força para conclusão deste trabalho.

# Resumo

Este texto é um exemplo, use como modelo para a formação do seu TG, não leve em consideração o conteúdo do texto aqui apresentado, trata-se apenas de um exemplo. O resumo deve ser composto de uma sequência de frases concisas e afirmativas. Inicie com uma frase significativa, explicando o tema e o problema abordado, a seguir ressalte o objetivo geral, as técnicas e ferramentas, os resultados e a conclusão. Lembre-se que o trabalho será inicialmente conhecido pelo resumo aqui escrito, portanto, ele deve ser fidedigno ao trabalho desenvolvido. Note que não há abertura de parágrafo. O conteúdo deve ser disposto em parágrafo único. O resumo não deve ser muito longo, tente escrever no máximo 250 palavras, o objetivo do resumo é mostrar para o leitor qual é o escopo do TG. A ABNT recomenda que os resumos sejam escritos no impessoal, assim como todo o texto do TG. Por exemplo, deve-se utilizar as seguintes expressões: “verifica-se que", “conclui-se que", "percebe-se que pelos testes", "é válido supor", etc. Não é adequado, dizer: "conforme vimos no item anterior". Diz-se: "conforme visto no item anterior", ou, em vez de "dissemos que", "foi dito que". No caso do TG em grupo, é permitido usar a primeira pessoa do plural. Não indique referências.

Palavras-chave: Resumo; Pesquisa; Escopo

# Lista de Figuras

[Figura 1: Caso de uso de alto nível. 17](#_Toc7547352)

[Figura 2: Arquitetura do sistema. 32](#_Toc7547353)

[Figura 3: Diagrama de classes. 35](#_Toc7547354)

[Figura 4: Modelo conceitual do Banco de Dados. 37](#_Toc7547355)

[Figura 5: Modelo lógico do Banco de Dados. 38](#_Toc7547356)

[Figura 6: Sequência para Cadastrar Patrimônio 39](#_Toc7547357)

[Figura 7: Sequência para Alterar Patrimônio 40](#_Toc7547358)

[Figura 8: Sequência para Realizar Baixa de Patrimônio 41](#_Toc7547359)

[Figura 9: Sequência para Exportar Excel 42](#_Toc7547360)

[Figura 10: Sequência para Cadastrar Localização 43](#_Toc7547361)

[Figura 11: Sequência para Alterar Localização 43](#_Toc7547362)

[Figura 12: Sequência para Excluir Localização 44](#_Toc7547363)

[Figura 13: Sequência para Cadastrar Aquisição 44](#_Toc7547364)

[Figura 14: Sequência para Alterar Aquisição 45](#_Toc7547365)

[Figura 15: Sequência para Excluir Aquisição 45](#_Toc7547366)

[Figura 16: Sequência para Cadastrar Usuário 46](#_Toc7547367)

[Figura 17: Sequência para Alterar Usuário 47](#_Toc7547368)

[Figura 18: Sequência para Excluir Usuário 47](#_Toc7547369)

[Figura 19: Diagrama de Pacotes 48](#_Toc7547370)

# Lista de Tabelas

**Índice**

[Resumo 1](#_Toc7956005)

[Lista de Figuras 2](#_Toc7956006)

[Lista de Tabelas 3](#_Toc7956007)

[1. Introdução 6](#_Toc7956008)

[2. Planejamento Inicial do Software 7](#_Toc7956009)

[2.1 Situação Atual 7](#_Toc7956010)

[2.1.1 Descrição da Situação Atual 7](#_Toc7956011)

[2.1.2 Problemas Encontrados 8](#_Toc7956012)

[3.2 Objetivos de Negócio do Projeto 9](#_Toc7956013)

[3.3 Restrições e Riscos do Projeto Atual 11](#_Toc7956014)

[3.3.1 Limitações Operacionais 11](#_Toc7956015)

[3.3.2 Considerações Legais 11](#_Toc7956016)

[3.3.3 Considerações de Hardware / Software / Rede 12](#_Toc7956017)

[3.3.4 Políticas Organizacionais 12](#_Toc7956018)

[3.3.5 Riscos 13](#_Toc7956019)

[4. Análise de Requisitos 14](#_Toc7956020)

[4.1 Descrição da técnica utilizada para levantamento dos requisitos 14](#_Toc7956021)

[4.2 Situação Proposta 14](#_Toc7956022)

[4.3 Requisitos Funcionais 15](#_Toc7956023)

[4.3.1 Diagrama de Casos de Uso e Descrição dos Casos de Uso 17](#_Toc7956024)

[4.4 Requisitos Não Funcionais 31](#_Toc7956025)

[5. Projeto Detalhado do Software 32](#_Toc7956026)

[5.1 Arquitetura da aplicação Atual 32](#_Toc7956027)

[5.2 Tecnologias utilizadas e APIs 33](#_Toc7956028)

[5.2.1 Tecnologias 33](#_Toc7956029)

[5.2.2 APIs Utilizadas 33](#_Toc7956030)

[5.3 Componentes do SW 34](#_Toc7956031)

[5.4 Diagrama de Classes 35](#_Toc7956032)

[5.5 Banco de Dados 36](#_Toc7956033)

[5.5.1 Modelo Conceitual 36](#_Toc7956034)

[5.5.2 Modelo Lógico 37](#_Toc7956035)

[5.6 Diagrama de Sequência 38](#_Toc7956036)

[5.6.1. Cadastro Patrimônio 38](#_Toc7956037)

[5.6.2. Alteração Patrimônio 39](#_Toc7956038)

[5.6.3. Baixa de Patrimônio 40](#_Toc7956039)

[5.6.4. Exportar Excel 41](#_Toc7956040)

[5.6.5. Cadastro de Localização 42](#_Toc7956041)

[5.6.6. Alteração de Localização 42](#_Toc7956042)

[5.6.7. Exclusão de Localização 43](#_Toc7956043)

[5.6.8. Cadastro Método de Aquisição 43](#_Toc7956044)

[5.6.9. Alteração Método de Aquisição 44](#_Toc7956045)

[5.6.10. Exclusão de Método de Aquisição 44](#_Toc7956046)

[5.6.11. Cadastro Usuário 45](#_Toc7956047)

[5.6.12. Alteração Usuário 46](#_Toc7956048)

[5.6.13. Exclusão de Usuário 46](#_Toc7956049)

[5.7 Diagrama Pacotes 47](#_Toc7956050)

[5.8 Diagrama Estado 48](#_Toc7956051)

[5.9 Interfaces com o usuário 49](#_Toc7956052)

[5.10 Relatórios 50](#_Toc7956053)

[6. Implementação 51](#_Toc7956054)

[7. Projeto de Teste 52](#_Toc7956055)

[8. Instalação do software 53](#_Toc7956056)

[9. Análise dos Resultados 54](#_Toc7956057)

[10. Conclusão 55](#_Toc7956058)

# Introdução

A Faculdade de Tecnologia de Sorocaba é uma das principais instituições de ensino de Sorocaba, foi a primeira escola pública de nível superior na cidade. A Instituição possui nove cursos de graduação presencial em tecnologia: Análise e Desenvolvimento de Sistemas, Eletrônica Automotiva, Fabricação Mecânica, Logística, Manufatura Avançada, Polímeros, Processos Metalúrgicos, Projetos Mecânicos e Sistemas Biomédicos. A Faculdade atende muitos alunos da região e possuí uma grande infraestrutura, assim existe um número considerável de patrimônios a serem controlados. O presente trabalho tem como objetivo desenvolver um software de controle de patrimônios, baseado na metodologia de trabalho e nas necessidades do setor de Patrimônio da Faculdade de Tecnologia de Sorocaba. Atualmente o setor de Patrimônio faz o gerenciamento das informações por meio de planilhas Excel, que são utilizadas em todos os processos, inclusive na comunicação com o Centro Paula Souza, que é a instituição mantenedora das Fatecs. Essa proposta foi definida em decorrência da necessidade de o setor possuir um controle mais automatizado, moderno e dinâmico das informações que são gerenciadas pela instituição.

O sistema irá prover funcionalidades para inclusão, alteração e exclusão de localização, meios de aquisição e patrimônio. Para cada patrimônio será possível realizar a baixa e a inclusão de uma imagem referente ao item. O sistema também realizará importação e exportação de planilhas, facilitando processos burocráticos exigidos pelo Centro Paula Souza.

O sistema será desenvolvido utilizando a metodologia ágil Scrum, com as seguintes ferramentas e tecnologias: o framework Spring da linguagem Java, que irá fornecer bibliotecas e funcionalidades que irão agilizar o desenvolvimento da API; o framework Angular da linguagem JavaScript, para proporcionar um desenvolvimento orientado a componentes para a parte visual do sistema; e o armazenamento das informações será realizada no banco de dados SQL Server. A arquitetura de software que será adotada para o desenvolvimento do sistema será a SOFEA (Arquitetura Front-End Orientada a Serviços).

# Planejamento Inicial do Software

## Situação Atual

No presente momento, o setor de Patrimônio da Faculdade de Tecnologia de Sorocaba guarda e administra seus dados por meio de uma planilha no Excel e, por essa razão os funcionários têm encontrado problemas para gerir esses dados, além de não poderem identificá-los de forma visual.

### Descrição da Situação Atual

A Faculdade conta com 15 Prédios distribuídos pelo campus, mais de 3000 alunos matriculados e 300 funcionários, assim possuí uma grande lista de patrimônios a controlar, com cerca de 8000 itens. Essa estrutura requer um gerenciamento eficaz dos patrimônios alocados na instituição.

A partir das dificuldades encontradas pelos funcionários, foi realizado um inventário com alunos voluntários do curso de logística. Esses alunos atualizaram a planilha de patrimônio, porém ao tentar adicionar fotos para os patrimônios, o uso de planilha se tornou inviável, uma vez que a inserção e utilização de fotos não foi possível em decorrência da grande massa de registros existentes. A proposta é que seja desenvolvido um software que importe a planilha, dê suporte a todas as operações hoje realizadas e que seja possível adicionar fotos de todos os itens de forma a não afetar o desempenho.

Atualmente, na planilha, é realizada a inserção de patrimônios, remoção de patrimônios quando há baixa patrimonial e a alteração da localização dos patrimônios quando são movidos.

### 2.1.2 Problemas Encontrados

Após entrevista com os funcionários do setor de patrimônio, foram identificados vários problemas decorrentes da utilização de um documento Excel para a gestão:

* Impossibilidade de associar imagens aos registros dos patrimônios;
* Falta de automação no processo de baixa patrimonial, pois é necessário gerar um outro arquivo Excel, manualmente, com todos os registros que irão passar pelo processo e, posteriormente, fazer a baixa manual dos registros no Excel;
* Baixa restrição de acesso e segurança das informações gerenciadas. Todos os dados são armazenados em um único arquivo Excel, com isso todas essas informações ficam vulneráveis a perdas e alterações indevidas. Essas condições podem gerar, na pior das hipóteses, a perda de dados inseridos ou até mesmo a perda de todo o documento de controle de patrimônios;
* Dados centralizados em um único arquivo Excel, o que não possibilita o trabalho simultâneo nativamente;
* A interface não é agradável ao usuário e gera desconforto no uso contínuo do sistema.

## Objetivos de Negócio do Projeto

Durante a etapa de planejamento foram definidos objetivos que tem por finalidade o aumento da eficiência, confiabilidade e da segurança dos processos executados pelo setor de controle de patrimônio, bem como oferecer aos funcionários um sistema amigável e de fácil aprendizado.

**Adição de foto aos registros dos Patrimônios**

O sistema irá fornecer este recurso que tem como objetivo facilitar a identificação dos patrimônios, com ele será possível adicionar uma foto do patrimônio ao seu respectivo registro no sistema.

**Suporte ao processo de baixa patrimonial**

O sistema irá possibilitar que o usuário crie o arquivo Excel de baixa patrimonial de forma automatizada selecionando os registros desejados em uma lista de patrimônios. O sistema também irá fornecer ao usuário a possibilidade de realizar o processo de baixa em si de forma automatizada, para isso, será necessário apenas que o usuário copie os valores do número de patrimônio do Excel gerado e cole no componente de entrada definido no sistema.

**Trabalho simultâneo sobre os dados gerenciados pelo sistema**

O sistema irá possibilitar que diversos funcionários do setor de patrimônio tenham acesso e possam gerenciar, simultaneamente, os dados armazenados no sistema, sem nenhum tipo de configuração prévia.

**Controle de acesso ao sistema de gerenciamento de patrimônio**

O sistema irá fornecer um método de controle de acesso seguro aos dados gerenciados pelo sistema por meio de um login e senha que serão somente cadastrados pelos administradores do setor de patrimônios. Assim, só os funcionários e pessoal autorizado poderão acessar o sistema.

**Armazenamento de dados seguro e *backup* periódico**

O sistema irá utilizar um sistema gerenciador de banco de dados atual e altamente seguro, impossibilitando o acesso indevido aos dados gerenciados e gerados na utilização do sistema de controle de patrimônio. Também serão realizados *backups* periódicos, garantindo assim que nenhum ou poucos dados sejam perdidos em caso de catástrofes ambientais, falhas de hardware e outras situações que venham a afetar o servidor utilizado.

**Interface moderna, de fácil utilização e aprendizado**

O sistema irá proporcionar aos usuários uma interface moderna, de uso agradável, simples e fácil de se aprender. Desenvolvida em um processo contínuo de prototipação juntamente com os funcionários do setor de patrimônio.

## 3.3 Restrições e Riscos do Projeto Atual

### 3.3.1 Limitações Operacionais

No servidor será necessário um sistema operacional com suporte ao ambiente Java 8, recomendamos a utilização do Windows Server ou uma das versões do Linux para servidor como Debian ou Ubuntu Server para a execução da camada de API e Backend do sistema. Para a execução do sistema na máquina de usuários é necessário o uso do Windows 7 ou versão mais atual, recomendamos que a configuração seja de pelo menos 8 GB de RAM e um processador i3 ou superior para uma utilização em conjunto com outros softwares instalados na máquina. Com relação ao Banco de dados será necessário a versão 2014 do SQL Server ou posterior e pelo menos 10 GB reservados para o armazenamento no servidor. Os dispositivos de entrada necessários serão mouse e teclado, também será possível a utilização por meio de uma tela *touchscreen* juntamente com o teclado virtual; os dispositivos de saída são monitores ou outros dispositivos de tela.

### 3.3.2 Considerações Legais

O sistema é um software licenciado por meio da GNU GPL. Sendo a licença para software livre mais utilizada no mundo, se baseando em quatro liberdades: (1) a liberdade de executar o programa, para qualquer propósito; (2) a liberdade de estudar como o programa funciona e adaptá-lo às suas necessidades; (3) a liberdade de redistribuir cópias de modo que se possa ajudar ao seu próximo; e (4) a liberdade de aperfeiçoar o programa, e liberar suas modificações, de modo que toda a comunidade se beneficie delas. Assim, o sistema está disponível ao acesso e uso dentro das diretrizes especificadas acima.

### 3.3.3 Considerações de Hardware / Software / Rede

As camadas de API, Backend e Banco de Dados do sistema serão executadas em um ambiente de Servidor com espaço de disco rígido mínimo de 10GB. Será necessário no mínimo 16GB de memória RAM para acomodar o sistema junto com outras aplicações com um bom desempenho. A camada de Front-End do sistema será executada em um ambiente Desktop de no mínimo 4 GB de RAM e processador i3 ou equivalente, também será possível a execução em *notebooks*, *smartphones* ou *tablets* com desempenho equivalente. Será necessário um outro servidor, em uma localização diferente do servidor utilizado pelo sistema, sendo de preferência na nuvem, para realização de *backups* periódicos.

### 3.3.4 Políticas Organizacionais

Para garantir o sigilo das informações armazenadas no banco de dados, o acesso ao sistema só será autorizado por meio de um *login* e senha previamente cadastrados pelo administrador responsável. É também recomendado a realização de *backups* diária e que ambos, servidor e computadores envolvidos na execução do sistema, estejam protegidos com antivírus e *firewall* sempre atualizados e configurados.

### 3.3.5 Riscos

Para que um projeto possa ser concluído no prazo estipulado é necessário um plano para caso ocorra alguma situação exista uma maneira de tratá-la e assim minimizar os danos no projeto.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **PLANO DE RISCOS** | | | | |
|
| **Risco** | **Probab (P)** | **Impacto (I)** | **RiscoTotal (P x I)** | **Tratamento** |
|
| Falta de funcionários capacitados | 2 | 5 | 10 | Realizar treinamentos em plataformas online como Udemy |
| Omissão por parte do entrevistado | 1 | 5 | 5 | Entrevistar outros funcionários que executem a mesma função |
| Mudanças constante nos processos | 1 | 5 | 5 | Utilizar metodologias ágeis |
| Falta de documentos | 3 | 3 | 9 | Solicitar documentação utilizada nos processos da empresa com antecedência |
| Problemas de Integração | 4 | 3 | 12 | Fazer uma apresentação e treinamento dos funcionários |
| Falta de energia | 2 | 4 | 8 | Contratação/Compra de geradores. |
| Sistema necessita de mais recursos do que esperado | 1 | 4 | 4 | Otimizar o sistema ou comprar equipamentos que disponibilizem esses recursos |
| Catástrofe Naturais | 1 | 5 | 5 | Servidor ter uma sala própria e *backups* guardados em lugares seguros e distintos. |
| Perda de dados | 1 | 5 | 5 | Fazer *backup* diariamente. |
| Incêndio | 1 | 5 | 5 | Sistema de proteção contra incêndio |

|  |  |
| --- | --- |
| **Probabilidade e Impacto** | |
| **Valor** | **Descrição** |
| 1 | Muito baixa |
| 2 | Baixa |
| 3 | Média |
| 4 | Alta |
| 5 | Muito alta |

# 4. Análise de Requisitos

## 4.1 Descrição da técnica utilizada para levantamento dos requisitos

Para o levantamento de requisitos, foram utilizadas as técnicas de entrevista, em que o usuário principal do sistema foi submetido a perguntas relativas ao sistema e a análise de uma planilha, em que havia todos os dados do sistema.

## 4.2 Situação Proposta

Está sendo proposto um novo sistema (desktop/web) no qual será possível diminuir a quantidade de ações repetitivas que o usuário teria que fazer, como por exemplo, escrever o mesmo método de aquisição várias vezes, escrever uma mesma localização para o patrimônio várias vezes, além de fazer a separação dos patrimônios ativos dos os patrimônios em baixa e possibilitar que os patrimônios tenham uma foto de identificação. O propósito do sistema é automatizar o meio de registrar os dados referentes ao patrimônio da instituição de modo que a adição, alteração ou remoção de algum patrimônio ocorra de forma simplificada.

## 4.3 Requisitos Funcionais

RF1 - Incluir Localização

Este requisito refere-se ao cadastro de novas localizações que serão atribuídas aos patrimônios.

RF2 – Alterar Localização

Este requisito refere-se a alterações das localizações já cadastradas.

RF3 – Desativar Localização

Este requisito refere-se à desativação de localizações não mais utilizadas pelos usuários.

RF4 – Incluir Patrimônio

Este requisito refere-se ao cadastro de novos patrimônios.

RF5 – Alterar Patrimônio

Este requisito refere-se a alterações dos patrimônios já cadastrados.

RF6 – Baixa Patrimônio

Este requisito refere-se à baixa nos patrimônios cadastrados.

RF7 - Incluir Método de Aquisição

Este requisito refere-se ao cadastro de novos modos de aquisição.

RF8 - Alterar Método de Aquisição

Este requisito refere-se a alterações dos modos de aquisição já cadastradas.

RF9 - Desativar Método de Aquisição

Este requisito refere-se à desativação de um modo de aquisição.

RF10 – Efetuar Login

Este requisito refere-se à efetuação de *Login* no sistema.

RF11 – Incluir Usuário

Este requisito refere-se ao cadastro de novos usuários no sistema.

RF12 – Alterar Usuário

Este requisito refere-se a alterações dos usuários já cadastrados no sistema.

RF13 – Desativar Usuário

Este requisito refere-se à desativação de usuários já cadastrados no sistema.

### 4.3.1 Diagrama de Casos de Uso e Descrição dos Casos de Uso

O diagrama de alto nível é utilizado para expor as funcionalidades do sistema de forma simples, de forma que ao visualizá-lo seja possível compreender como ele irá funcionar.

Figura 1: Caso de uso de alto nível.

Uma imagem contendo texto, mapa, interior, mesa

Descrição gerada automaticamente

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF1: INCLUIR LOCALIZAÇÃO | |
| **Ator Principal** | USUÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** |  | |
| **Pós-Condição** | Nova localização armazenada no sistema | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – O usuário acessa a tela “Localização” e seleciona a opção “Novo” | |  |
|  | | 2 – Apresenta a tela “Nova Localização” com os campos “Código” e “Descrição” e os botões “Salvar” e “Cancelar”. O campo “Código” é preenchido automaticamente com um determinado código criado pelo sistema para a identificação da localização. |
| 3 – O usuário preenche o campo “Descrição” e clica no botão “Salvar” | |  |
|  | | 4 – Armazena no banco os dados da nova localização |
|  | | 5 – Retorna mensagem “Operação Realizada Com Sucesso” |

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF2: ALTERAR LOCALIZAÇÃO | |
| **Ator Principal** | USUÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** | Localização estar cadastrada no sistema | |
| **Pós-Condição** | Alterações da localização estarem armazenadas no sistema | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – O usuário acessa a tela “Localização” | |  |
|  | | 2 – Lista todas as localizações já registradas no sistema |
| 3 – O usuário seleciona a localização para alteração e clica no botão “Alterar” | |  |
|  | | 4 – Apresenta a tela “Alterar Localização” com os campos “Código” e “Descrição” preenchidos de acordo com a seleção do usuário. Exibe os botões “Salvar” e “Cancelar”. |
| 5 – O usuário realiza as alterações e clica no botão “Salvar” | |  |
|  | | 6 – Armazena no banco os dados alterados da localização |
|  | | 7 – Retorna mensagem “Operação Realizada Com Sucesso” |

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF3:DESATIVAR LOCALIZAÇÃO | |
| **Ator Principal** | USUÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** | Localização estar cadastrada no sistema | |
| **Pós-Condição** | Localização estar registrada como “Desativada” no sistema | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – O usuário acessa a tela “Localização” | |  |
|  | | 2 – Lista todas as localizações já registradas no sistema |
| 3 – O usuário seleciona a localização para desativação e clica no botão “Alterar” | |  |
|  | | 4 – Apresenta a tela “Alterar Localização” com os campos “Código” e “Descrição” preenchidos de acordo com a seleção do usuário e o campo “Status” com a opção “Ativo”. Exibe os botões “Salvar” e “Cancelar”. |
| 5 – O usuário seleciona a opção “Ativo” para desativar e clica no botão “Salvar” | |  |
|  | | 6 – Armazena no banco os dados alterados da localização |
|  | | 7 – Retorna mensagem “Operação Realizada Com Sucesso” |

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF4:INCLUIR PATRIMÔNIO | |
| **Ator Principal** | USUÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** |  | |
| **Pós-Condição** | Novo patrimônio armazenado no sistema | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – O usuário acessa a tela “Patrimônio” e seleciona o botão “Novo” | |  |
|  | | 2 – Apresenta a tela “Novo Patrimônio” com os campos “Localização”, “Número do Patrimônio”, “Número Processo de Aquisição”, “Descrição”, “Nota Fiscal”, “Marca”, “Modelo”, “Número de Série”, “Informações Complementares”, “Valor”, “Modalidade de Aquisição” e “Imagem”. Os campos “Localização” e “Modalidade de Aquisição” são preenchidos com os dados contidos no banco. Exibe também os botões “Selecionar”, “Salvar” e “Cancelar” |
| 3 – O usuário preenche os campos “Número do Patrimônio”, “Número Processo de Aquisição”, “Descrição”, “Nota Fiscal”, “Marca”, “Modelo”, “Número de Série”, “Informações Complementares” e “Valor”. Seleciona os dados pros campos “Localização” e “Modalidade de Aquisição”. Clica no botão “Selecionar” e seleciona a imagem desejada. Clica no botão “Salvar” | |  |
|  | | 4 – Armazena no banco os dados do novo patrimônio |
|  | | 5 – Retorna mensagem “Operação Realizada Com Sucesso” |

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF5:ALTERAR PATRIMÔNIO | |
| **Ator Principal** | USUÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** | Patrimônio estar cadastrado no sistema | |
| **Pós-Condição** | Alterações do patrimônio estarem armazenadas no sistema | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – O usuário acessa a tela “Patrimônio” | |  |
|  | | 2 – Lista todos os patrimônios já registrados no sistema |
| 3 – O usuário seleciona o patrimônio para alteração e clica no botão “Alterar” | |  |
|  | | 4 – Apresenta a tela “Alterar Patrimônio” com os campos “Localização”, “Número do Patrimônio”, “Número Processo de Aquisição”, “Descrição”, “Nota Fiscal”, “Marca”, “Modelo”, “Número de Série”, “Informações Complementares”, “Valor”, “Modalidade de Aquisição” e “Imagem”. Os campos “Localização” e “Modalidade de Aquisição” são preenchidos com os dados contidos no banco. Exibe também os botões “Selecionar”, “Salvar” e “Cancelar” |
| 5 – O usuário realiza as alterações e clica no botão “Salvar” | |  |
|  | | 6 – Armazena no banco os dados alterados da localização |
|  | | 7 – Retorna mensagem “Operação Realizada Com Sucesso” |

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF6:BAIXA PATRIMÔNIO | |
| **Ator Principal** | USUÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** | Patrimônio estar cadastrado no sistema | |
| **Pós-Condição** | Patrimônio estar armazenado em estado de baixa. | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – O usuário acessa a tela “Baixa em Patrimônio” | |  |
|  | | 2 – Lista todos os patrimônios já registrados no sistema. Exibe as opções de consulta e os botões “Exportar Excel” e “Realizar Baixa” |
| 3 – O usuário seleciona o(s) patrimônio(s) para dar baixa e clica no botão “Exportar Excel” | |  |
|  | | 4 – Exporta todos os dados do(s) patrimônio(s) selecionado(s) para um arquivo Excel. |
| 5 – O usuário seleciona o botão “Realizar Baixa” | |  |
|  | | 6 – Apresenta a tela “Realizar Baixa” com um campo de inserção de texto e os botões “Cancelar” e “Realizar Baixa” |
| 7 – O usuário insere um número de patrimônio por linha e clica no botão “Realizar Baixa” | |  |
|  | | 8– Realiza a leitura em cada linha individualmente, busca cada número de patrimônio no banco de dados e efetua a baixa em cada patrimônio individualmente, colocando-os em estado de baixa |
|  | | 9– Armazena no banco os dados alterados dos patrimônios |
|  | | 10 – Retorna mensagem “Operação Realizada Com Sucesso” |

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF7: INCLUIR MÉTODO DE AQUISIÇÃO | |
| **Ator Principal** | USUÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** |  | |
| **Pós-Condição** | Nova aquisição armazenada no sistema | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – O usuário acessa a tela “Método de Aquisição” e seleciona a opção “Novo” | |  |
|  | | 2 – Apresenta a tela “Novo Método de Aquisição” com os campos “Código” e “Descrição” e os botões “Salvar” e “Cancelar”. O campo “Código” é preenchido automaticamente com um determinado código criado pelo sistema para a identificação do método de aquisição |
| 3 – O usuário preenche o campo “Descrição” e clica no botão “Salvar” | |  |
|  | | 4 – Armazena no banco os dados do novo método de aquisição |
|  | | 5 – Retorna mensagem “Operação Realizada Com Sucesso” |

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF8: ALTERAR MÉTODO DE AQUISIÇÃO | |
| **Ator Principal** | USUÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** | Método de aquisição estar cadastrada no sistema | |
| **Pós-Condição** | Alterações do método de aquisição estarem armazenadas no sistema | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – O usuário acessa a tela “Método de Aquisição” | |  |
|  | | 2 – Lista todos os métodos de aquisição já registradas no sistema |
| 3 – O usuário seleciona o método de aquisição para alteração e clica no botão “Alterar” | |  |
|  | | 4 – Apresenta a tela “Alterar Método de Aquisição” com os campos “Código” e “Descrição” preenchidos de acordo com a seleção do usuário. Exibe os botões “Salvar” e “Cancelar”. |
| 5 – O usuário realiza as alterações e clica no botão “Salvar” | |  |
|  | | 6 – Armazena no banco os dados do novo método de aquisição |
|  | | 7 – Retorna mensagem “Operação Realizada Com Sucesso” |

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF9:DESATIVAR MÉTODO DEAQUISIÇÃO | |
| **Ator Principal** | USUÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** | Método de aquisição estar cadastrada no sistema | |
| **Pós-Condição** | Método de aquisição estar registrado como “Desativada” no sistema | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – O usuário acessa a tela “Método de Aquisição” | |  |
|  | | 2 – Lista todos os métodos de aquisição já registradas no sistema |
| 3 – O usuário seleciona o método de aquisição para desativação e clica no botão “Alterar” | |  |
|  | | 4 – Apresenta a tela “Alterar Método de Aquisição” com os campos “Código” e “Descrição” preenchidos de acordo com a seleção do usuário e o campo “Status” com a opção “Ativo”. Exibe os botões “Salvar” e “Cancelar”. |
| 5 – O usuário seleciona a opção “Ativo” para desativar e clica no botão “Salvar” | |  |
|  | | 6 – Armazena no banco os dados do novo método de aquisição |
|  | | 7 – Retorna mensagem “Operação Realizada Com Sucesso” |

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF10: EFETUAR *LOGIN* | |
| **Ator Principal** | USUÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** | Usuário estar cadastrado no sistema | |
| **Pós-Condição** | Usuário pode ter acesso ao sistema de acordo com as disponibilidades do seu nível de hierarquia | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – O usuário abre o sistema | |  |
|  | | 2 – Apresenta a tela “*Login*” com os campos “Usuário” e “Senha” e o botão “Entrar” |
| 3 – O usuário preenche os campos “Usuário” e “Senha” e clica no botão “Entrar” | |  |
|  | | 4 – Efetua uma busca no banco de dados para verificar o usuário e senha e o nível de acesso do usuário |
|  | | 5 – Apresenta a tela “Início” e deixa apenas as funcionalidades permitidas de acordo com o nível de hierarquia do usuário ativadas |

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF11:INCLUIR USUÁRIO | |
| **Ator Principal** | USUÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** | Estar conectado ao sistema com uma conta de nível de hierarquia mais alto (administrador) | |
| **Pós-Condição** | Novo usuário armazenado no sistema | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – O usuário acessa a tela “Consulta de Usuários” e seleciona o botão “Novo” | |  |
|  | | 2 – Apresenta a tela “Cadastro de Usuário” com os campos “Nome”, “Usuário”, “Senha” e um combo box “Nível de Hierarquia”. Exibe também os botões “Salvar” e “Cancelar” |
| 3 – O usuário preenche os campos “Nome”, “Usuário” e “Senha” e seleciona o nível de hierarquia desejado para o novo usuário. Clica no botão “Salvar” | |  |
|  | | 4 – Armazena no banco os dados do novo usuário |
|  | | 5 – Retorna mensagem “Operação Realizada Com Sucesso” |

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF11:ALTERAR USUÁRIO | |
| **Ator Principal** | USUÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** | Estar conectado ao sistema com uma conta de nível de hierarquia mais alto (administrador) | |
| **Pós-Condição** | Alterações do usuário armazenadas no sistema | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – O usuário acessa a tela “Consulta de Usuários” e seleciona um usuário e clica no botão “Alterar” | |  |
|  | | 2 – Lista todos os usuários já registrados no sistema |
| 3 – O usuário seleciona um usuário e clica no botão “Alterar” | |  |
|  | | 2 – Apresenta a tela “Alterar Usuário” com os campos “Nome”, “Usuário”, “Senha” e um combo box “Nível de Hierarquia” preenchidos de acordo com os dados salvos no banco. Exibe também os botões “Salvar” e “Cancelar” |
| 3 – O usuário efetua as alterações desejadas e clica no botão “Salvar” | |  |
|  | | 4 – Armazena no banco os novos dados do usuário |
|  | | 5 – Retorna mensagem “Operação Realizada Com Sucesso” |

|  |  |  |
| --- | --- | --- |
| **Caso de Uso** | RF12:DESATIVAR USUÁRIO | |
| **Ator Principal** | USUÁRIO | |
| **Ator Secundário** |  | |
| **Pré-Condição** | Estar conectado ao sistema com uma conta de nível de hierarquia mais alto (administrador) | |
| **Pós-Condição** | Usuário armazenado em estado de “Desativado” no sistema | |
| **Ações do Ator** | | **Ações do Sistema** |
| 1 – O usuário acessa a tela “Consulta de Usuários” e seleciona um usuário e clica no botão “Alterar” | |  |
|  | | 2 – Apresenta a tela “Alterar Usuário” com os campos “Nome”, “Usuário”, “Senha” e um combo box “Nível de Hierarquia” preenchidos de acordo com os dados salvos no banco. Exibe também os botões “Salvar” e “Cancelar” e um check box com a opção “Ativo”. |
| 3 – O usuário seleciona o check box para deixar em estado desativado e clica no botão “Salvar” | |  |
|  | | 4 – Armazena no banco os novos dados do usuário |
|  | | 5 – Retorna mensagem “Operação Realizada Com Sucesso” |

## 4.4 Requisitos Não Funcionais

RNF1 – Restrição de Acesso

O usuário precisará de um *login* e uma senha previamente cadastrados pelos administradores para ter acesso ao sistema.

RNF2 – Sistema Operacional

Sistema operacional com suporte ao ambiente java 8.

RNF3 – Rede

Será necessária conexão de rede entre a Api, Backend, banco de dados e a máquina do usuário.

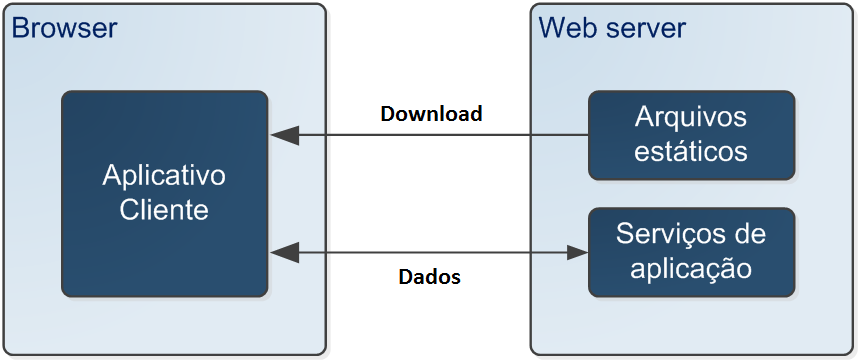
# 5. Projeto Detalhado do Software

Para o desenvolvimento do projeto foi utilizada a metodologia ágil Scrum tomando como base para geração de documentação os principais Diagramas da UML.

## 5.1 Arquitetura da aplicação Atual

Na arquitetura do sistema foi utilizada a arquitetura SOFEA, que pode ser vista abaixo na figura 2.

Figura 2: Arquitetura do sistema.



## 5.2 Tecnologias utilizadas

As tecnologias utilizadas para o desenvolvimento do aplicativo serão relacionadas a seguir.

Front-End: Angular

**Descrição do Angular**

Back-End: Spring

Spring é um framework de código aberto criado por Rod Johnson em meados de 2002, e apresentado no seu livro Expert One-on-One: JEE Design and Development. Foi criado com o intuito simplificar a programação em Java, possibilitando construir aplicações que antes só era possível utilizando EJB’s (Enterprise JavaBeans).

O Spring atualmente possui diversos módulos como Spring Data (trata da persistência), Spring Security (trata da segurança da aplicação) entre outros módulos. Mas o *core* pode ser utilizado em qualquer aplicação Java, as principais funcionalidades são a injeção de dependência (CDI) e a programação orientada a aspectos (AOP), cabe ao desenvolvedor dizer ao Spring o que quer usar. O que faz dele uma poderosa ferramenta, pois não existe a necessidade de se arrastar todas as ferramentas do framework para criar uma aplicação simples.

Banco de Dados: SQL Server

O Microsoft SQL Server é um sistema gerenciador de Banco de dados relacional (SGBD) desenvolvido pela Microsoft. Como um Banco de dados, é um produto de software cuja principal função é a de armazenar e recuperar dados solicitados por outras aplicações de software, seja aqueles no mesmo computador ou aqueles em execução em outro computador através de uma rede (incluindo a Internet).

## 5.3 Componentes do SW

Descrever os módulos da aplicação.

## 5.4 Diagrama de Classes

Figura 3: Diagrama de classes.

Uma imagem contendo captura de tela

Descrição gerada automaticamente

## 5.5 Banco de Dados

O banco de dados utilizado no sistema será Microsoft SQL Server 2014 da Microsoft Corporation, versão- 12.0.2269.0

### 5.5.1 Modelo Conceitual

Figura 4: Modelo conceitual do Banco de Dados.

Uma imagem contendo texto

Descrição gerada automaticamente

### 5.5.2 Modelo Lógico

Figura 5: Modelo lógico do Banco de Dados.

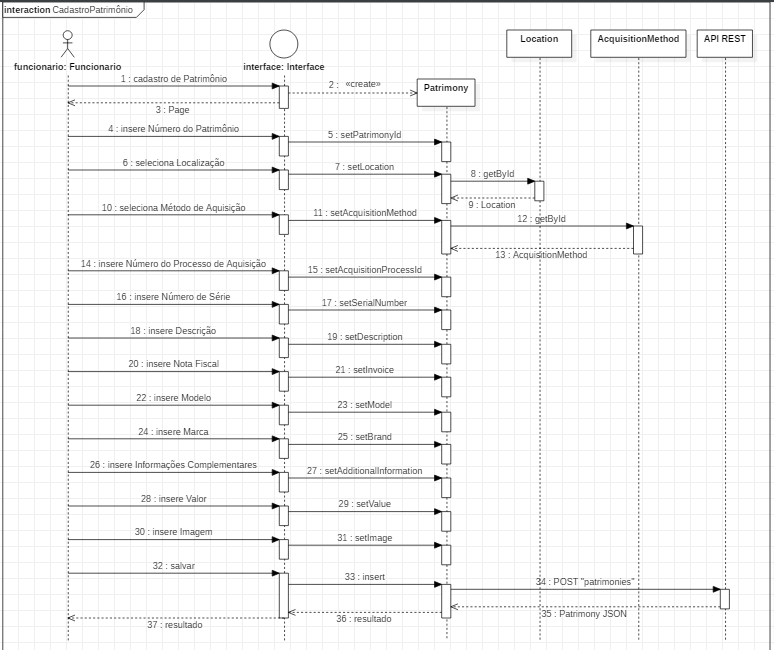
Uma imagem contendo captura de tela

Descrição gerada automaticamente

## 5.6 Diagrama de Sequência

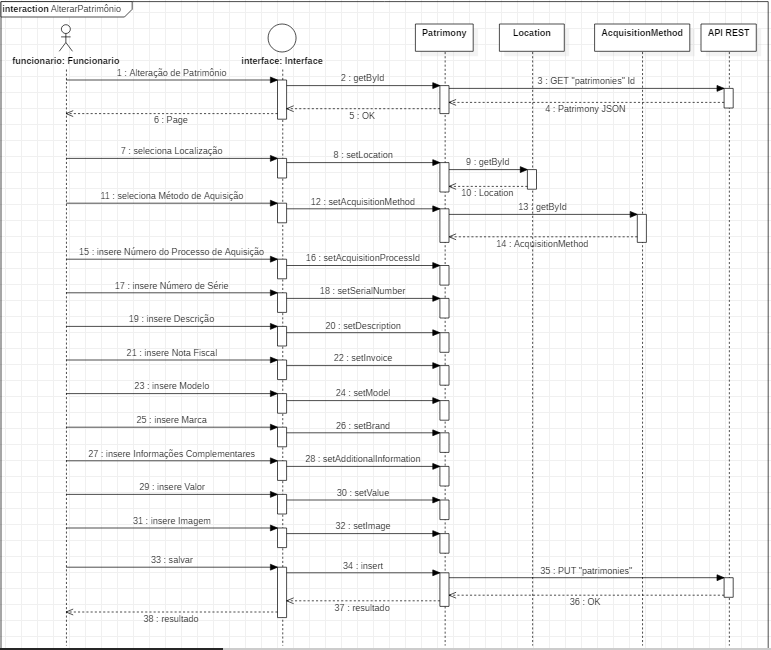
### 5.6.1. Cadastro Patrimônio

Figura 6: Sequência para Cadastrar Patrimônio



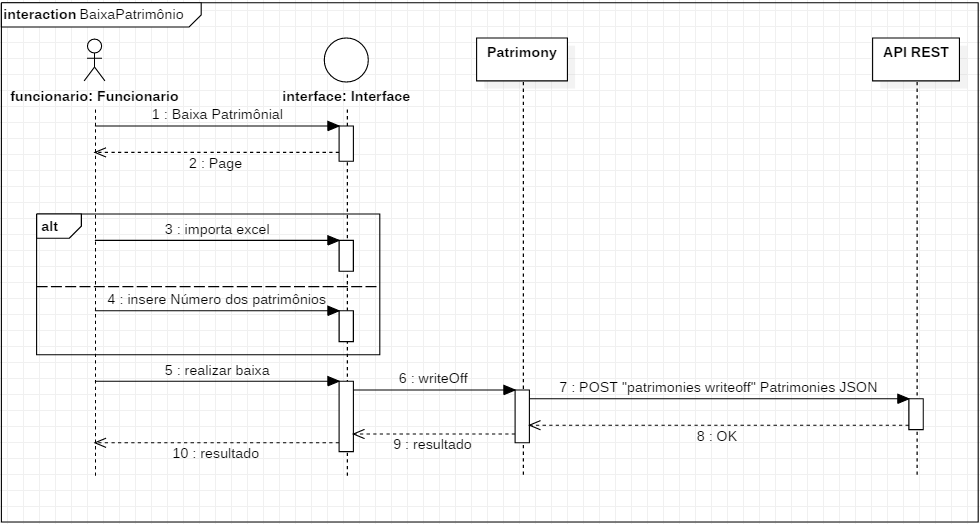
### 5.6.2. Alteração Patrimônio

Figura 7: Sequência para Alterar Patrimônio



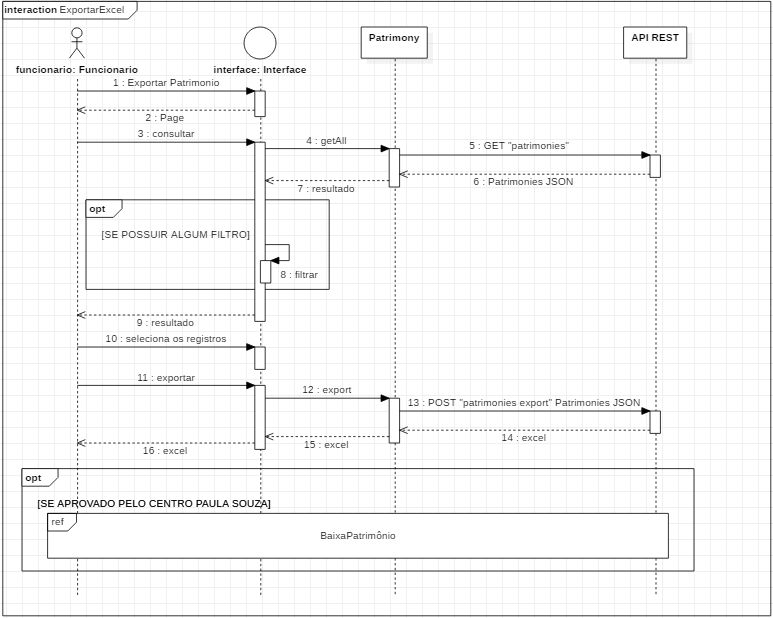
### 5.6.3. Baixa de Patrimônio

Figura 8: Sequência para Realizar Baixa de Patrimônio



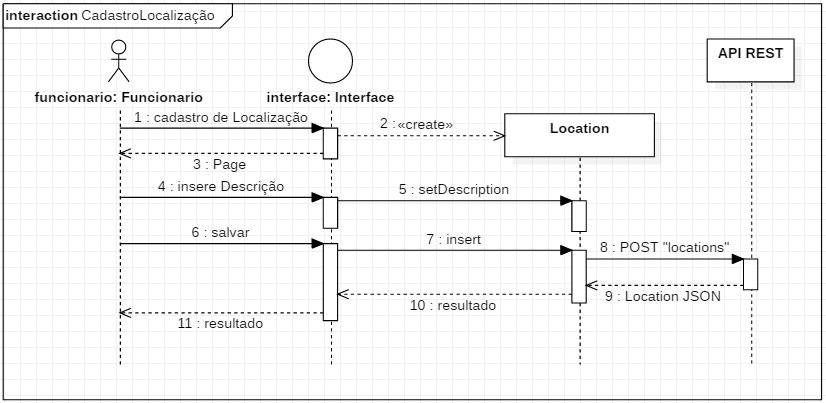
### 5.6.4. Exportar Excel

Figura 9: Sequência para Exportar Excel



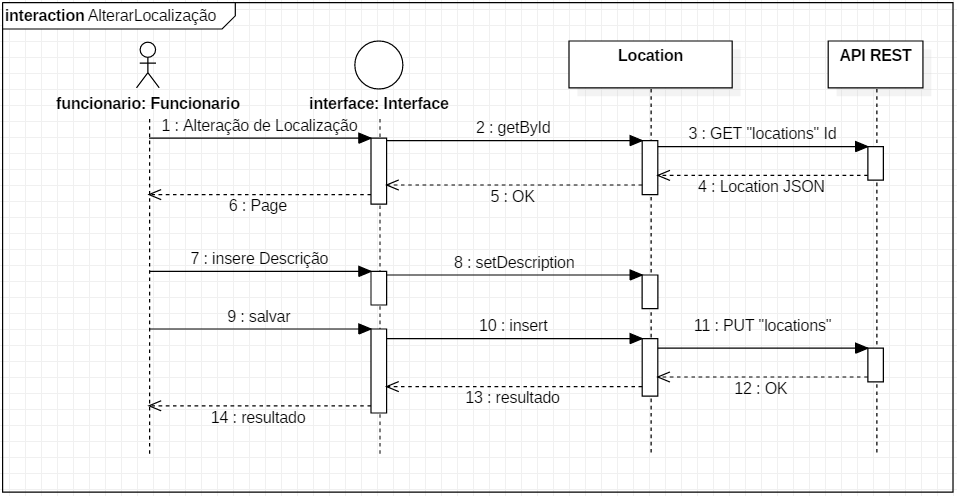
### 5.6.5. Cadastro de Localização

Figura 10: Sequência para Cadastrar Localização



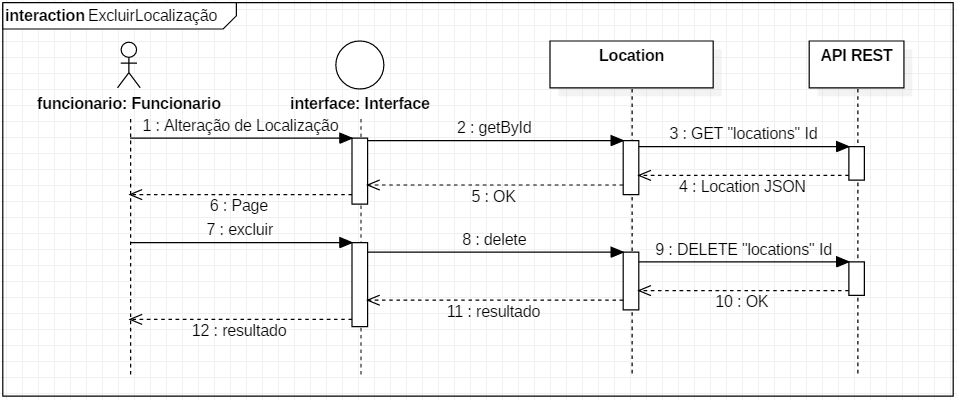
### 5.6.6. Alteração de Localização

Figura 11: Sequência para Alterar Localização



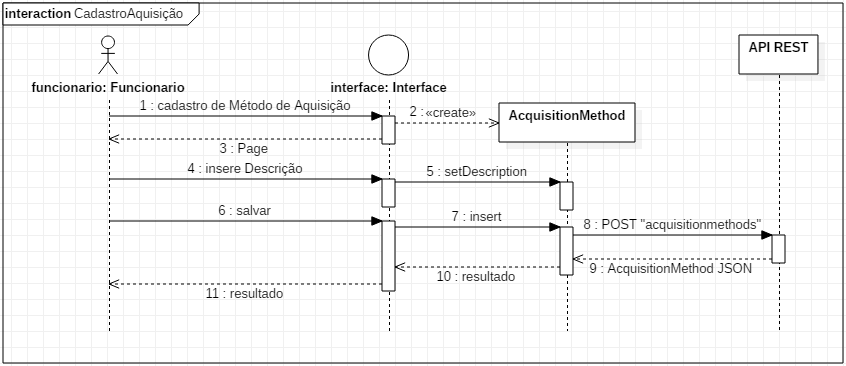
### 5.6.7. Exclusão de Localização

Figura 12: Sequência para Excluir Localização



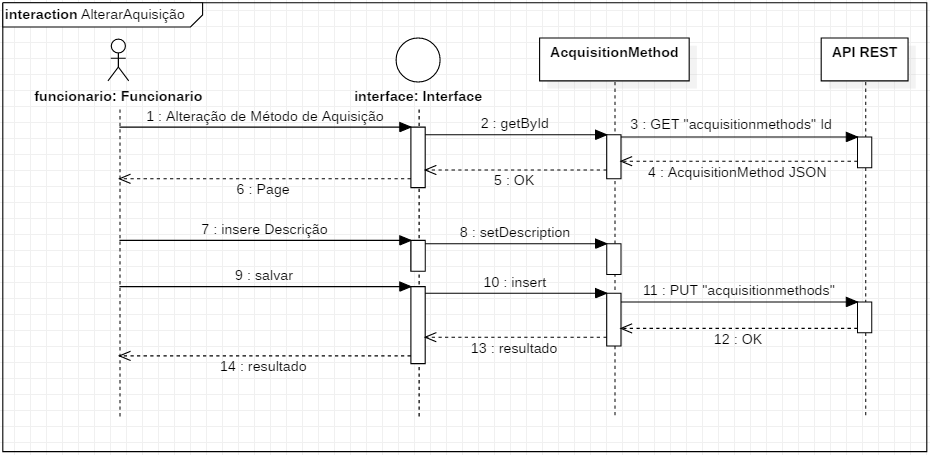
### 5.6.8. Cadastro Método de Aquisição

Figura 13: Sequência para Cadastrar Aquisição



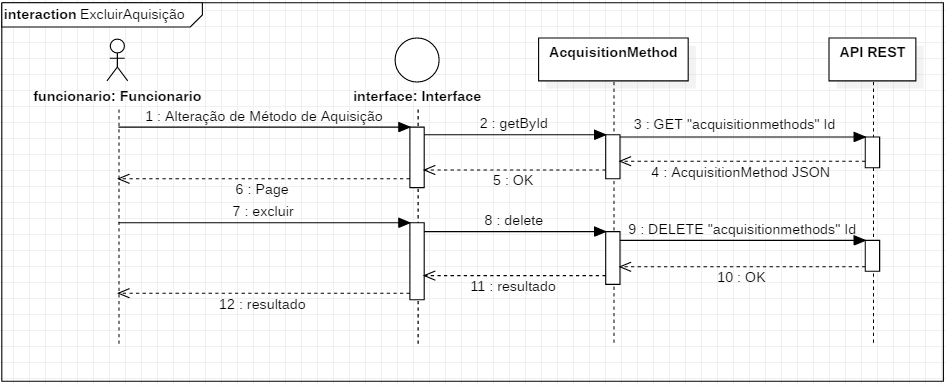
### 5.6.9. Alteração Método de Aquisição

Figura 14: Sequência para Alterar Aquisição



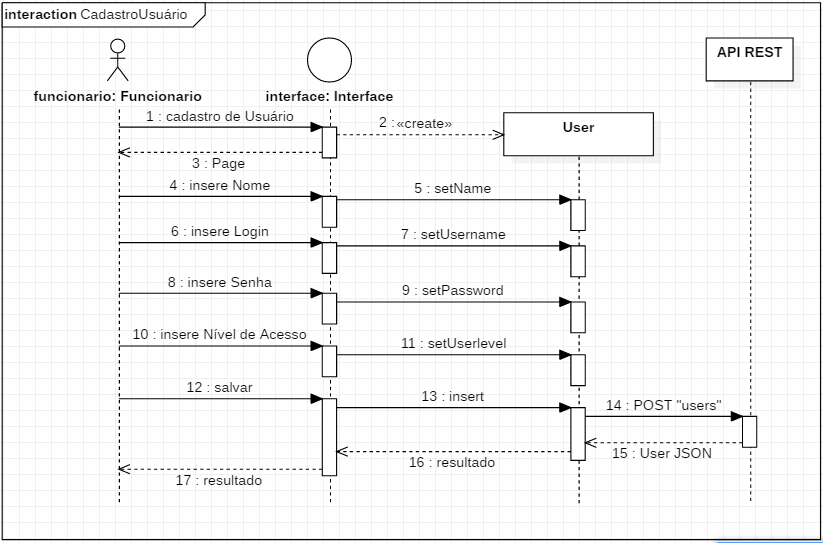
### 5.6.10. Exclusão de Método de Aquisição

Figura 15: Sequência para Excluir Aquisição



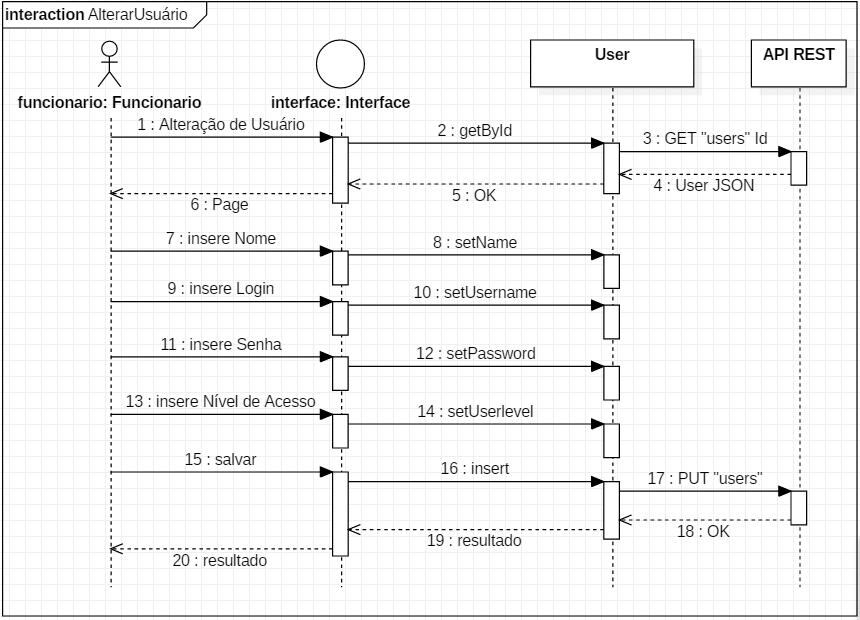
### 5.6.11. Cadastro Usuário

Figura 16: Sequência para Cadastrar Usuário



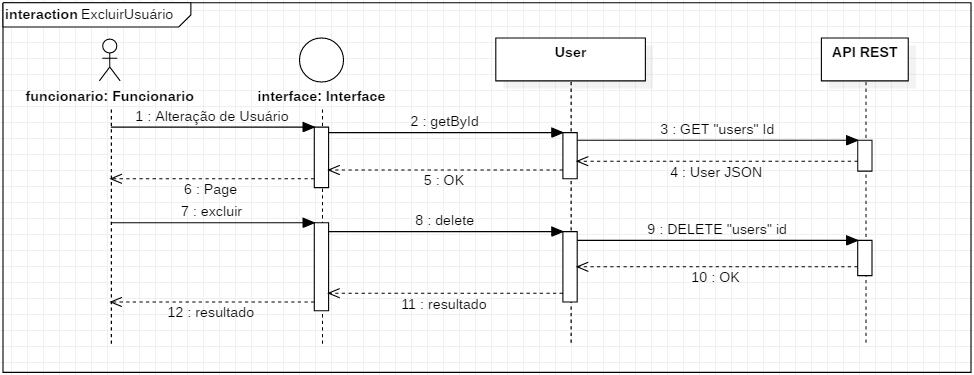
### 5.6.12. Alteração Usuário

Figura 17: Sequência para Alterar Usuário



### 5.6.13. Exclusão de Usuário

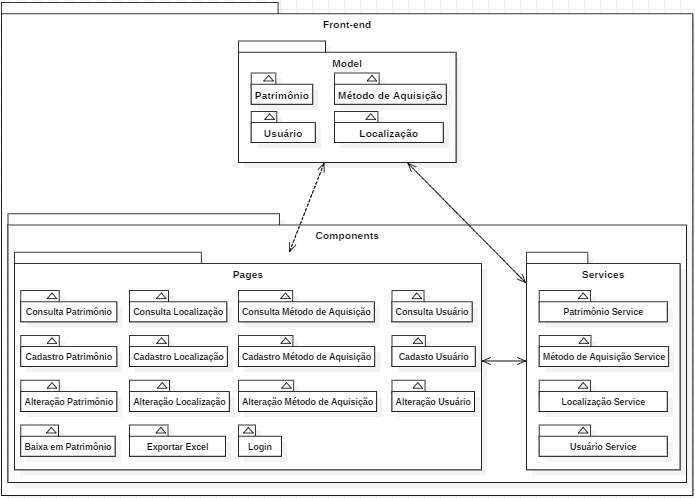
Figura 18: Sequência para Excluir Usuário



## 5.7 Diagrama Pacotes

O diagrama de pacotes permite uma abordagem mais ampla do sistema, sendo possível visualizar as comunicações entre os pacotes ou subsistemas. Segundo McConnell (2005) “Particularmente nesse nível são as regras a respeito de como os vários subsistemas podem se comunicar”, com isso pode-se observar que ao separar as classes em pacote é possível ter um controle de como as comunicações acontecerão quando o sistema estiver em produção.

Figura 19: Diagrama de Pacotes



## 5.8 Diagrama Estado

O diagrama de estado é tipicamente um complemento para a descrição das classes. Este diagrama mostra todos os estados possíveis que objetos de uma certa classe podem se encontrar, e mostra também quais são os eventos dos sistemas que provocam tais mudanças. Ele captura todo o ciclo de vida dos objetos, subsistemas e sistemas. Ele mostra os estados que um objeto pode possuir e como os eventos afetam estes estados ao passar do tempo.

## 5.9 Interfaces com o usuário

Apresentar aqui as Interfaces com o usuário. Pode ser *printscreen* das telas ou layout.

## 5.10 Relatórios

Descrever os relatórios disponibilizados.

# 6. Implementação

O código fonte pode ser encontrado no repositório do projeto no GitHub, nos links:

<https://github.com/gabrielfelicidade/patrimonio-back-end>

<https://github.com/gabrielfelicidade/patrimonio-front-end>

# 7. Projeto de Teste

No desenvolvimento do projeto serão utilizadas técnicas para testar processos, encontrar e corrigir erros, como por exemplo o teste unitário que permite ao desenvolvedor corrigir pequenos trechos de código. Assim como serão utilizadas ferramentas de Debug que permitem observar mais a fundo o modo que a lógica foi implementada no código. (McConnell, 2005).

# 8. Instalação do software

Fornece informações sobre a instalação do software desenvolvido, assim como dos softwares complementares a serem instalados para o funcionamento do sistema.

# 9. Conclusão

Este item é muito importante. Faz o fechamento, concluindo as ideias. Esta etapa sintetiza todo o trabalho realizado e fornece uma resposta para a questão apresentada. Pode também levantar hipóteses e refletir sobre cada objetivo proposto.

A conclusão deverá apresentar um resumo de tudo o que foi feito. Poderão ser inseridos argumentos que mostrem quais objetivos foram atingidos e os resultados obtidos.

Referências

PRESSMAN, Roger. S. Engenharia de software: uma abordagem profissional. 7. ed. Porto Alegre: AMGH, 2011. 780p.

PRESSMAN, Roger S; MAXIM, Bruce. Engenharia de Software: Uma abordagem profissional. 8. ed. AMGH, 2016.

**Anexos**

É opcional. Documentos agregados à obra para fins de comprovação de dados ou ilustração.

**Glossário**

É um item opcional. Trata-se de uma listagem que contém as palavras ou termos técnicos desconhecidos utilizados no texto, com seus significados. A lista deve ser em ordem alfabética.

**Exemplo:**

**SGBD** – Sistema Gerneciador de Banco de Dados. Software que gerencia e proporciona o armazenamento de dados, permitindo consultas aos dados armazenados e garantindo sua integridade.

**Sistemas de Informação Gerencial** ou **ERP** – **E**nterprise **R**esource **P**lanning ou software de planejamento de recursos empresariais. É um software que procura integrar todas as áreas da empresa, desde o chão de fábrica até a alta administração, procurando otimizar processos e garantir confiabilidade das informações.

**Workflow** – Software que procura gerenciar e descrever o fluxo de dados entre as tarefas e processos da organização.

**Apêndice**

É opcional – São documentos agregados à obra para fins de apoio à argumentação. Nesta parte são incluídos os questionários, entrevistas, tabulação de dados, etc.

**REFERÊNCIAS**

**Borges, Daniella A. Franceschinelli.**  Material da disciplina de Engenharia de Software III , Fatec Sorocaba, 2017

**IFSC,2018 -** Dicas para escrita de texto cientifico. Disponível em **:** <https://wiki.sj.ifsc.edu.br/wiki/index.php/Dicas_para_escrita_de_texto_cient%C3%ADfico> Acesso em: 25/04/2018

**Medeiros, Ernani Sales de.** Desenvolvendo Software com UML. Makron Books – São Paulo, 2004

**Modelo Scrum.** SCRUMstudy™ - Guia SBOK™ - 2016.

**Munhoz, Levi Rodrigues**. Material da disciplina de Engenharia de Software II, Fatec Sorocaba, 2017

**Normas ABNT.** Disponível em [https://www.normaseregras.com/normas-abnt/](https://www.normaseregras.com/normas-abnt/%20%20)  Acesso em: 17/04/2018

**Sommerville, Ian*.*** Engenharia de Software. Ed. Addison Wesley - São Paulo, 2003.