**<Nome da Empresa>**

# Sistema de Reserva Especificação dos Requisitos

**Versão 1.0**

**Histórico da Revisão**

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| 29/10/2019 | 1.0 | Levantamento de Requisitos. | Amanda Simões, Daniel Dantas, Fagne Farias e Igor José. |
| 30/10/2019 | 1.0 | Modelagem do diagrama entidade relacionamento. | Amanda Simões e Daniel Dantas. |
| 30/10/2019 | 1.0 | Prototipação das tela de cadastro e login. | Igor José. |
| 31/10/2019 | 1.0 | Definição das tecnologias a serem utilizadas. | Amanda Simões, Daniel Dantas, Fagne Farias e Igor José. |
| 01/11/2019 | 1.0 | Prototipação da tela de inicio. | Fagne Farias e Igor José. |
| 01/11/2019 | 1.0 | Iniciou desenvolvimento usando Node JS. | Daniel Dantas. |
| 01/11/2019 | 1.0 | HTML, CSS e JS da tela de cadastro e login concluidos. | Fagne Farias. |
| 02/11/2019 | 1.0 | Inicio da documentação. | Amanda Simões. |
| 02/11/2019 | 1.0 | Modificações nas rotas para teste e criação de model. | Daniel Dantas. |
| 03/11/2019 | 1.0 | Mudanças de rotas para controles. | Daniel Dantas |
| 03/11/2019 | 1.0 | Inserção dos Casos de Uso(UseCase) e requisitos funcionais. | Amanda Simões. |
| 05/11/2019 | 1.0 | Criação de funcionalidades do sistema. | Daniel Dantas. |
| 06/11/2019 | 1.0 | Ajustes nas telas de cadastro e login e adição da tela de inicio. | Fagne Farias e Igor José. |
| 08/11/2019 | 1.0 | Ajuste nas telas e adição de prototipação. | Amanda Simões, Fagne Farias e Igor José. |
| 08/11/2019 | 1.0 | Inicialização da API rest. | Daniel Dantas. |
| 09/11/2019 | 1.0 | Ajustes nos Casos de Uso e inserção do Diagrama de Classe. | Amanda Simões, Daniel Dantas e Igor José. |
| 09/11/2019 | 1.0 | Finalização dos Casos de Uso e Diagrama de Classe. Inserção e finalização do Diagrama de Objeto. | Amanda Simões. |
| 09/11/2019 | 1.0 | Ajustes nas telas. | Amanda Simões e Igor José. |
| 09/11/2019 | 1.0 | Funcionalidade de autenticação implementada. | Daniel Dantas. |
| 09/11/2019 | 1.0 | Inserção da tela de “Recuperação de Senhas”. | Fagne Farias. |
| 09/11/2019 | 1.0 | Linkagem da API rest com a tela de “Recuperação de Senhas” e linkagem das reservas com o front-end. | Daniel Dantas. |
| 10/11/2019 | 1.0 | Adição de envio de email quando usuario se cadastra. | Daniel Dantas. |
| 10/11/2019 | 1.0 | Finalização das telas e Documentação. | Amanda Simões e Igor José. |

|  |  |  |
| --- | --- | --- |
|  | **Índice** |  |
| **1. INTRODUÇÃO** |  | **5** |
| 1.1 OBJETIVOS |  | 5 |
| 1.2 PÚBLICO ALVO |  | 5 |
| 1.3 ESCOPO DO PRODUTO |  | 5 |
| 1.4 DEFINIÇÕES, ACRÔNIMOS E ABREVIAÇÕES |  | 5 |
| **2. REQUISITOS FUNCIONAIS** |  | **6** |
| **3. REQUISITOS NÃO-FUNCIONAIS** |  | **7** |
| **4. CASOS DE USO** |  | **8** |
| **5. DIAGRAMA DE CLASSE** |  | **11** |
| **6. DIAGRAMA DE OBJETO** |  | **12** |

1. **Introdução**

O Sistema de Reservas foi desenvolvimendo com o intuito de resolver o problema das reservas no IFPB – Campus Cajazeiras, no qual eram tudo feito no papel que acabava trazendo demora na realização de reservas de sala, laboratório e ambientes como piscina, quadra, campo de futebol e auditório.

## Objetivos

Este software tem como objetivo resolver o problema das reservas de sala, laboratório, datashow e ambiente como piscina, auditório, campo de futebol e a quadra no IFPB – Campus Cajazeiras, ajudando tanto na eficiência como na rapidez das reservas.

## Público Alvo

Temos como público alvo os professores e alunos do Campus.

## Escopo do Produto

O Sistema de Reservas tem as seguintes funcionalidades: O Admin(Administrador geral) poderá adicionar salas, laboratórios, DataShow e ambiente(Piscina, Quadra, Campo de Futebol e Auditório) e o Usuário, alunos e professores poderão realizar reservas, mas apenas se estiverem cadastrados no sistema. O sistema não permite que um aluno realize uma reserva sem uma autorização de um professor responsável, então o mesmo deve enviar uma solicitação para o professor, no qual o mesmo decide se defere o pedido ou não.

## 1.4 Definições, acrônimos e abreviações

**Prioridade dos requisitos:**

**Essencial:** indica que o requisito é imprescindível para o funcionamento do sistema. Requisitos essenciais devem ser implementados desde as primeiras iterações do desenvolvimento construção do sistema.

**Importante:** indica que o requisito não é essencial para o funcionamento do sistema, contudo seu funcionamento, sem implementação do requisito, se torna insatisfatório. Requisitos importantes devem ser implantados o mais rápido possível, porém não impede que apenas parte do sistema seja implantada.

**Desejável:** indica que o requisito não compromete as funcionalidades básicas do sistema, podendo funcionar de forma satisfatória sem ele. Requisitos desejáveis podem ser implantados por último, sem comprometer o funcionamento do sistema.

# 2. Requisitos Funcionais

**RF\_01:** Cadastro de Usuário – Cadastra professores e alunos no sistema sendo que ambos devem informar sua matrícula, um e-mail válido e uma senha. Se for professor a lista de disciplinas que o mesmo ministra também será cadastrado, se for aluno será informado qual curso.

Prioridade: [X] Essencial [   ] Importante [ ] Desejável

**RF\_02:** Login de Usuário – Usuário deve inserir matrícula e senha para poder logar no sistema.

Prioridade:  [X] Essencial [   ] Importante  [ ] Desejável

**RF\_03:** Cadastrar Sala – Admin deve cadastrar salas para serem reservadas. Deve ser cadastrado uma sigla para cada sala como por exemplo Sala01, no qual sempre que uma nova sala for cadastrada, ela deve ter o nome “Sala” no início e um número distinto sequencial na frente. Deve ser cadastrado também o bloco onde se encontra a sala e a capacidade de alunos.

Prioridade:  [X] Essencial [   ] Importante  [ ] Desejável

**RF\_04:** Cadastrar Laboratório – Admin deve cadastrar laboratórios para serem reservados. Deve ser cadastrado uma sigla para cada laboratório como por exemplo LabXX01, no qual sempre que um novo laboratório for cadastrado, ele deve ter o nome “Lab” no início, no lugar do XX colocar a especialidade do laboratório como por exemplo LabInfo01 ou LabBio01, e um número distinto sequencial na frente. Deve ser cadastrado também o bloco onde se encontra o laboratório e a capacidade de alunos.

Prioridade:  [X] Essencial [   ] Importante  [ ] Desejável

**RF\_05:** Cadastrar Ambiente – Admin deve cadastrar ambiente para serem reservados. Ambientes esses: Piscina, Ginásio, Campo de Futebol e Auditório. Deve ser cadastrado o nome do ambiente como por exemplo “Piscina”, Bloco no qual o mesmo se encontra e alguma observação como por exemplo capacidade ou qual roupa adequada deve ser usada naquele ambiente, como por exemplo “Touca de Banho” no ambiente “Piscina”.

Prioridade:  [X] Essencial [   ] Importante  [ ] Desejável

**RF\_06:** Cadastrar DataShow – Admin deve cadastrar DataShow para serem reservados. Deve ser cadastrado uma sigla para cada DataShow como por exemplo “DS01” sendo o número distinto e sequencial, deve também ser cadastrado alguma observação, se tiver, como por exemplo “Incluso com cabo HDMI”.

Prioridade:  [X] Essencial [   ] Importante  [ ] Desejável

**RF\_07:** Remover Sala – Admin poderá remover salas.

Prioridade:  [ ] Essencial [X] Importante  [ ] Desejável

**RF\_08:** Remover Laboratório – Admin poderá remover laboratórios.

Prioridade:  [ ] Essencial [X] Importante  [ ] Desejável

**RF\_09:** Remover Ambiente – Admin poderá remover ambiente.

Prioridade:  [ ] Essencial [X] Importante  [ ] Desejável

**RF\_10:** Remover Data Show – Admin poderá remover DataShow.

Prioridade:  [ ] Essencial [X] Importante  [ ] Desejável

**RF\_11:** Pedido de Autorização – Usuário aluno só pode realizar uma reserva com autorização de um professor, então o mesmo envia um pedido de autorização para um professor que também ficará responsável pela reserva.

Prioridade:  [ ] Essencial [X] Importante  [ ] Desejável

**RF\_12:** Autorização – Usuário professor pode autorizar ou não o requerimento do aluno.

Prioridade:  [ ] Essencial [X] Importante  [ ] Desejável

**RF\_13:** Histórico de Reservas – Usuário poderá visualizar todas as reservas podendo filtra por mês ou sala.

Prioridade:  [ ] Essencial [X] Importante  [ ] Desejável

**RF\_14:** Um usuário pode remover suas informações do sistema, caso resolva deixar de usar o mesmo.

Prioridade: [   ]Essencial [ ]Importante [X] Desejável

**RF\_15:** Um usuário pode atualizar suas informações do sistema, a qualquer momento

Prioridade: [   ]Essencial [ ]Importante [X] Desejável

**RF\_16:** O usuário poderá alterar o seu email ou senha.

Prioridade: [X]Essencial [   ]Importante [ ] Desejável

# Requisitos Não-Funcionais

**RNF\_01**: Segurança – Validação dos dados, e garantia de que as informações não vão se perder no meio das requisições, e garantir que os dados não vão cair na mão de terceiros.

Prioridade: [ X ] Essencial [ ] Importante [ ] Desejável

**RNF\_02**: Compatibilidade - O Sistema deve ser compatível para os seguintes navegadores: Chrome, Mozila e Opera.

Prioridade:  [ ] Essencial [X] Importante [   ] Desejável

# Casos de Uso

# UC\_1: Cadastro e Login

## Cenário típico

* 1. Passo 1: Usuario informa seus dados para se cadastrar.
  2. Passo 2: Os dados são Validados.
  3. Passo 3: Usuario é cadastrado.
  4. Passo 4: O usuario pode realizar o login.
  5. Passo 5: Usuario informa matricula e senha.
  6. Passo 6: Os dados são Validados.
  7. Passo 7: Usuario é redirecionado para a página inicial.

## Cenários alternativos

1.a Usuario já tem cadastro.

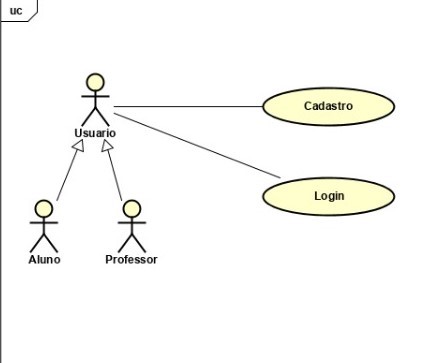
1. Retorna ao passo 4 do fluxo principal.

2.a Dados Invalidos.

1. Retorna ao passo 1 do fluxo principal.

6.a Dados Invalidos.

1. Retorna ao passo 4 do fluxo principal.



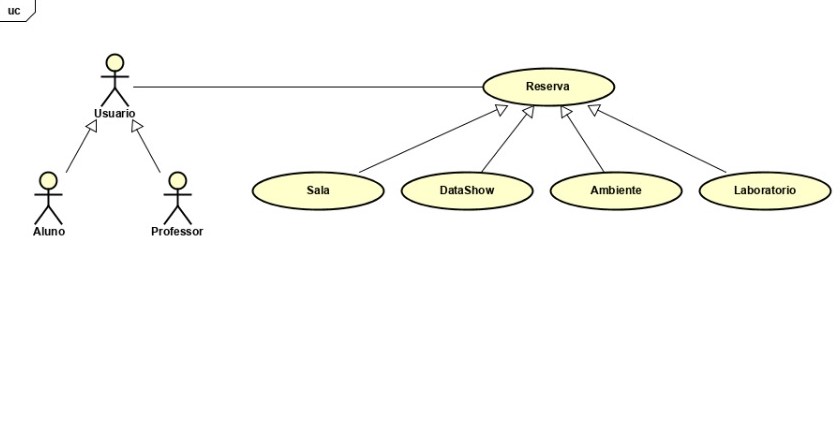
# UC\_1: Reserva

## Cenário típico

1. Passo 1: Usuario é redirecionado para a página principal.
2. Passo 2: Usuario inicia uma reserva.
3. Passo 3: Informa os dados.
4. Passo 4: Os dados são Validados.
5. Passo 5: Reserva Realizada com Sucesso.

## Cenários alternativos

* 1. Usuario pode cancelar a reserva.
     1. Retorna para o passo 1 do fluxo principal.
  2. Dados Invalidos.
     1. Retorna ao passo 2.



# UC\_1: Cadastro de Espaço

## Cenário típico

1. Passo 1: Admin informa dados para o login.
2. Passo 2: Os dados são validados.
3. Passo 3: Admin é redirecionado para a pagina inicial.
4. Passo 4: Admin inicia cadastro de espaço(Sala, laboratorio, ambiente ou datashow).
5. Passo 5: Informa os dados.
6. Passo 6: Os dados são validados.
7. Passo 7: Espaço cadastrado.

## Cenários alternativos

2.a Dados invalidos.

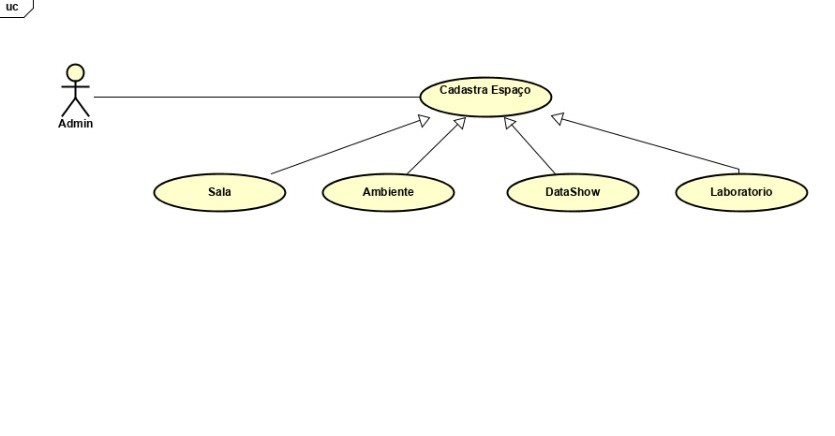
1. Retorna para o passo 1 do fluxo principal.

4.a Admin cancela o cadastro.

1. Retorna para o passo 3 do fluxo principal.

6.a Dados invalidos.

1. Retorna para o passo 4 do fluxo principal.



# 5. Diagrama de Classe

## 

## O seguinte diagrama representa as funcionalidades do sistemas, no qual usuario pode ser tanto aluno como professor e ambos podem realizar reservar, mas reserva só existe se existir usuarios. Pode ser reservado espaço(Sala e Laboratório), DataShow e Ambiente(Piscina, Auditório, Ginásio e Campo de Futebol, mas para eles existirem o sistema precisa necessariamente de um Administrador que fica responsavel por cadastra-los e por último não se pode existir usuario sem ter um Administrador, pois isso implica que não há ninguém para cadastrar eapaço, ambiente e datashow, logo não haverá reservas.

# Diagrama de Objeto

## 

O seguinte diagrama é uma represetação do diagrama de classe só que com instâncias de objetos para a simulação de como como os dados seriam inseridos.