

**Sistema de Gerenciamento**  
**Cliente: *Hostel***

**DOCUMENTO FINAL DO PROJETO**

**Gabriel Faria Stivanin 35188**

**Gabriel Rocha Pessoa 31325**

**Igor de Oliveira Gregorio 33630**

**Juliana Thais de Oliveira 33966**

**Leonardo Martins Souza 2016004210**

**Patricia Ferreira de Souza 2017015844**

**Revisões do Documento**

Revisões são melhoramentos na estrutura do documento e também no seu conteúdo. O objetivo primário desta tabela é a fácil identificação da versão do documento. Toda modificação no documento deve constar nesta tabela.

|  |  |  |
| --- | --- | --- |
| **Data** | **Versão** | **Descrição** |
| 15/05/2021 | 1.0 | Inserção dos Requisitos do Sistema e sua modelagem |
| 24/06/2021 | 1.1 | Renomeação dos CRUDS e sequências de acordo com as definições da SPRINT 1. |

Índice

[1.](#_heading=h.30j0zll) Introdução 4

[1.1](#_heading=h.1fob9te) Convenções, termos e abreviações 4

[*1.1.1*](#_heading=h.3znysh7) *Identificação dos Requisitos* 4

[1.1.2](#_heading=h.2et92p0) Prioridades dos Requisitos 4

[2.](#_heading=h.tyjcwt) Visão geral do Produto/serviço 5

[2.1](#_heading=h.3dy6vkm) Abrangência e sistemas relacionados 5

[2.2](#_heading=h.1t3h5sf) Descrição do cliente 6

[2.3](#_heading=h.4d34og8) Descrição dos usuários 6

[3.](#_heading=h.3rdcrjn) Estrutura Analítica do projeto (EAP) 7

[4.](#_heading=h.26in1rg) Modelagem do processo de negócio 7

[5.](#_heading=h.1ksv4uv) Requisitos funcionais de sistema (RFS) 10

[6.](#_heading=h.2jxsxqh) Requisitos Não Funcionais (RNF) 14

[7.](#_heading=h.z337ya) Modelo Conceitual do Banco de Dados 14

8[.](#_heading=h.1y810tw) Arquitetura da Aplicação 15

# Introdução

Este documento especifica para os desenvolvedores os requisitos de um sistema de informação, cuja finalidade é auxiliar funcionários e donos de Hostels a acompanharem a alocação de leitos e quartos nos Hostels.

Esta seção fornece as informações necessárias para fazer um bom uso deste documento, explicitando as convenções que foram adotadas no texto e as seções deste documento de requisitos de sistemas. As demais seções apresentam a especificação do **Sistema de Gerenciamento de Hostel** e estão organizadas como descrito abaixo:

* **Seção 2 – Visão geral do Produto**: apresenta uma visão geral do produto/serviço, caracterizando qual é o seu escopo e descrevendo seus usuários.
* **Seção 3 – Modelagem do Processo:** modelagem de todos os processos da aplicação **Sistema de Gerenciamento de Hostel**
* **Seção 4 - Requisitos funcionais:** lista e descreve os requisitos funcionais do produto/serviço, especificando seus objetivos, funcionalidades, atores e prioridades.
* **Seção 5 - Requisitos não funcionais**: especifica todos os requisitos não funcionais do produto/serviço.
* **Seção 6 - Modelo Conceitual do Banco de Dados:** modelagem do banco de dados da aplicação
* **Seção 7 – Modelo Lógico do Banco de Dados:** Disposição lógica do banco de dados
* **Seção 8 – Arquitetura da Aplicação:** Arquitetura simplificada e modelo de implantação do sistema
* **Seção 9 – Planejamento de Versões:** Definição dos requisitos que serão entregues na primeira e na última versão.

## **Convenções, termos e abreviações**

A correta interpretação deste documento exige o conhecimento de algumas convenções e termos específicos, que são descritos a seguir.

### ***Identificação dos Requisitos***

Por convenção, a referência a requisitos é feita através do identificador do requisito, conforme o esquema abaixo:

[Identificador de tipo de requisito. Identificador do requisito]

O identificador de tipo de requisito pode ser:

* RF – Requisito funcional
* RNF – Requisito não-funcional

Identificador do requisito é um número, criado sequencialmente, que determina que aquele requisito é único para um determinado tipo de requisito.

Ex: RF001, RF002, RNF001, RNF002.

### **Prioridades dos Requisitos**

Para estabelecer a prioridade dos requisitos foram adotadas as denominações “essencial”, “importante” e “desejável”.

* **Essencial** é o requisito sem o qual o sistema não entra em funcionamento. Requisitos essenciais são requisitos imprescindíveis, que têm que ser implementados impreterivelmente.
* **Importante** é o requisito sem o qual o sistema entra em funcionamento, mas de forma não satisfatória. Requisitos importantes devem ser implementados, mas, se não forem, o sistema poderá ser implantado e usado mesmo assim.
* **Desejável** é o requisito que não compromete as funcionalidades básicas do sistema, isto é, o sistema pode funcionar de forma satisfatória sem ele. Requisitos desejáveis são requisitos que podem ser deixados para versões posteriores do sistema, caso não haja tempo hábil para implementá-los na versão que está sendo especificada.

# Visão geral do Produto/serviço

Em uma sociedade cada vez mais ágil e tecnológica, na qual os trabalhos demoravam meses no passado são entregues em dias, e mudanças constantes no dia-a-dia, surge um novo desafio: organizar-se de maneira rápida e eficiente surfando na onda do *marketing* digital.

Controlar todas as contas e as receitas está se tornando um problema cada vez mais comum que pode ser resolvido com a tecnologia. O maior problema enfrentado pelos Hotels hoje é que, os *softwares* disponíveis no mercado não atendem ao estilo de estadia disponibilizado por essas empresas, que visam o aluguel de camas/leitos e não o de quarto, como em Hotéis. A partir disto, este grupo propõe-se a resolver o problema baseado nas seguintes dores:

1. Não ter um *software* que alugue apenas camas, independente do gênero.
2. A impossibilidade do administrador saber em tempo real um mapa completo de quais leitos estão alugados, por quanto tempo e quando estarão liberados novamente.
3. A dificuldade da recepcionista em lançar todos esses dados de maneira manual, sendo passível de erros (como duas reservas para mesmo leito ou nenhuma reserva).

A solução dessas dores visa facilitar os lançamentos e o gerenciamento do Hostel.

## **Abrangência e sistemas relacionados**

A solução desenvolvida é uma aplicação com dois usuários, sendo um para o administrador e outro para as recepcionistas. Em relação ao administrador, terá funcionalidades como controle financeiro, o lançamento de contas, fluxo de caixa e relatórios de estadias, receitas e despesas, além das funcionalidades contidas no perfil da recepcionista. O perfil das recepcionistas poderão lançar as estadias, lançar o fluxo de caixa, cancelamento de reservas e cadastro de hóspedes, leitos e quartos.

## Descrição do cliente

Os clientes que utilizarão o ***Hostel da Galera*** serão donos de Hostels e seus funcionários, que acessarão o sistema para o gerenciamento das estadias. Como descrito na seção 2 os administradores poderão enxergar um mapa mensal das reservas e proporcionar um histórico do cliente e das reservas de forma geral, auxiliando nas tomadas de decisão futuras.

## Descrição dos usuários

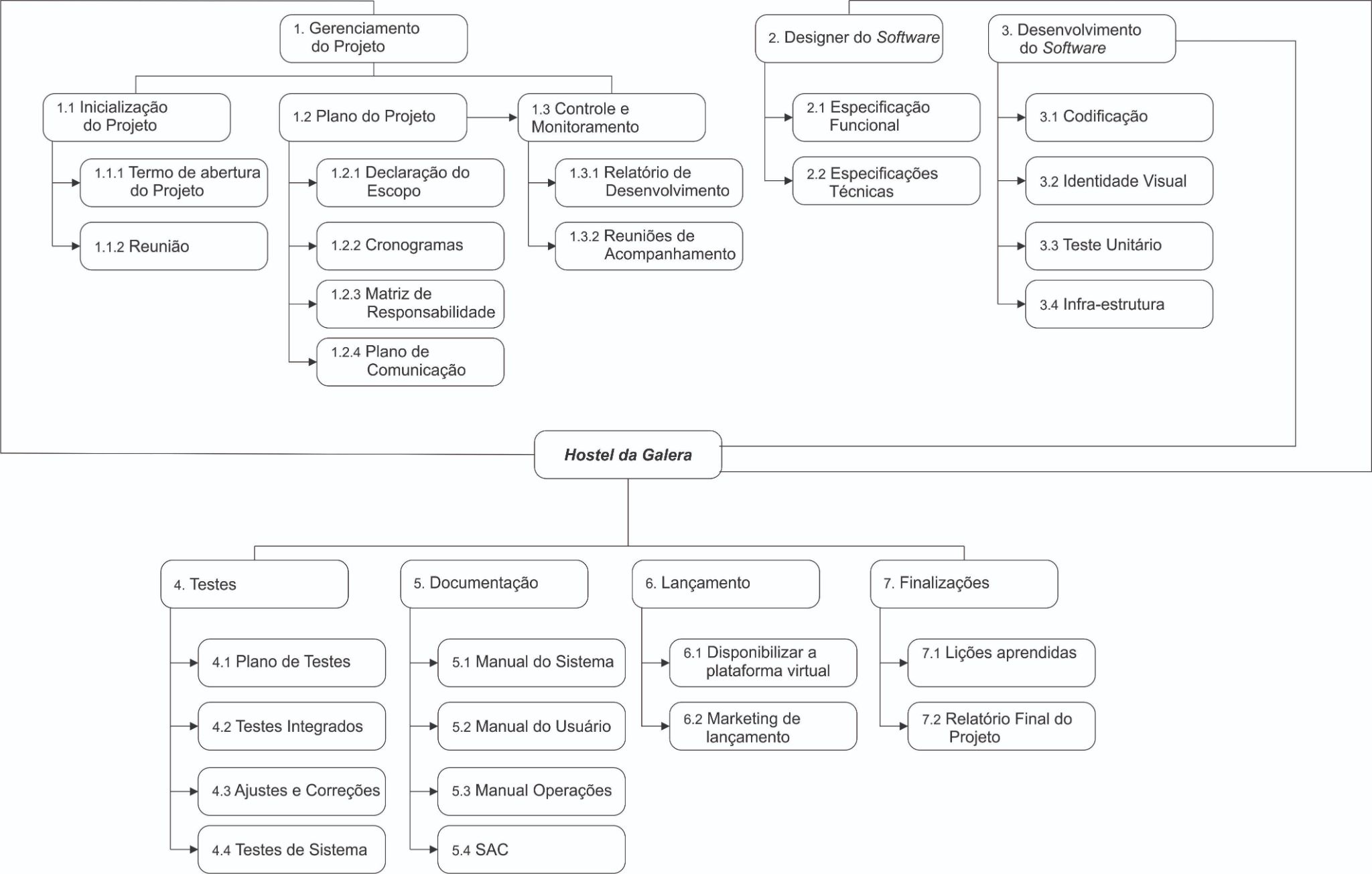
O sistema terá dois tipos de usuário:

* Administrador: será o usuário com o perfil de administrador do sistema.
* Recepcionista: Terá acesso limitado ao sistema, podendo fazer o cadastro de hóspedes, cadastro de leitos, reservas e lançamento financeiro.

**OBS:** O Administrador poderá executar todas as funções do sistema.

# Estrutura Analítica do projeto (EAP)

Esta seção pretende descrever os processos realmente necessários para a execução do projeto, desmembrando as fases e facilitando a realização das tarefas. A Figura 1 mostra esse processo.



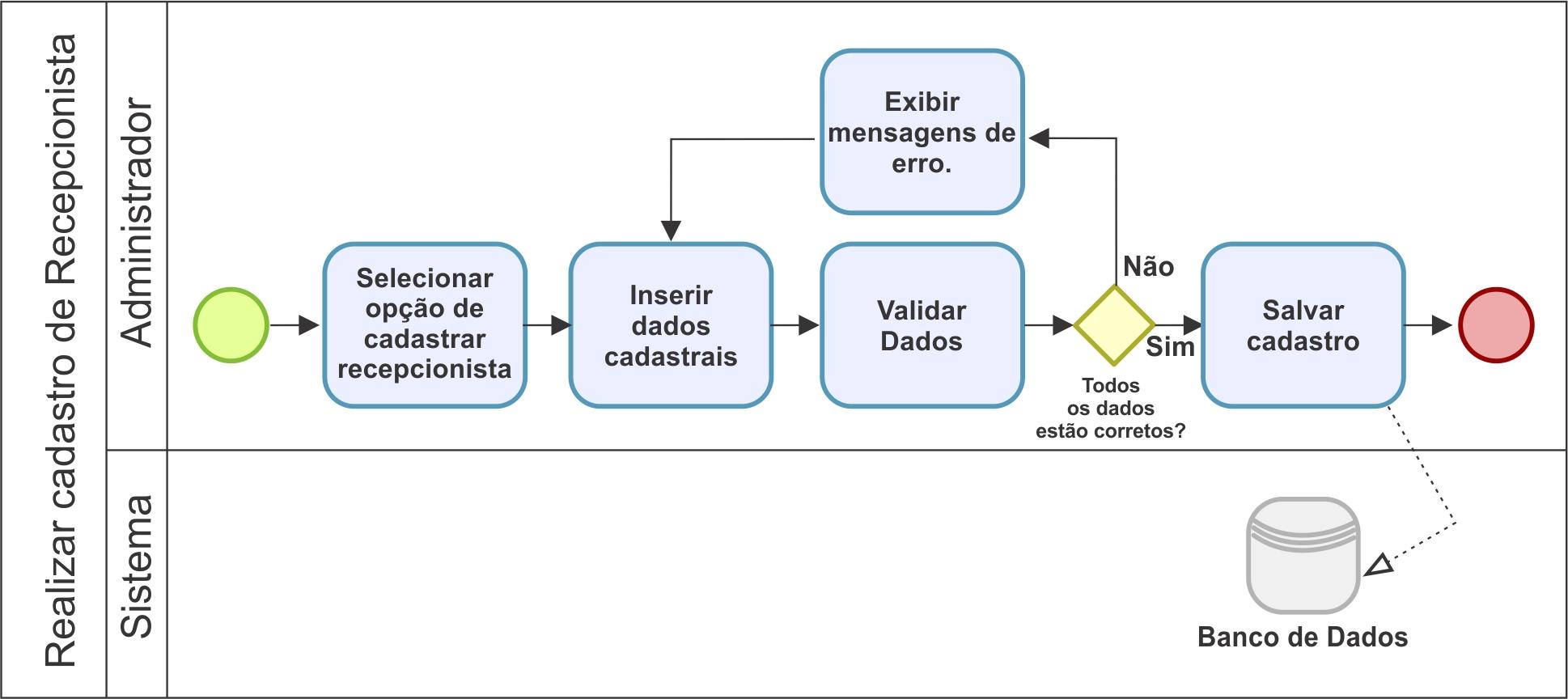
*Figura 1 – Modelagem EAP*

# Modelagem do processo de negócio

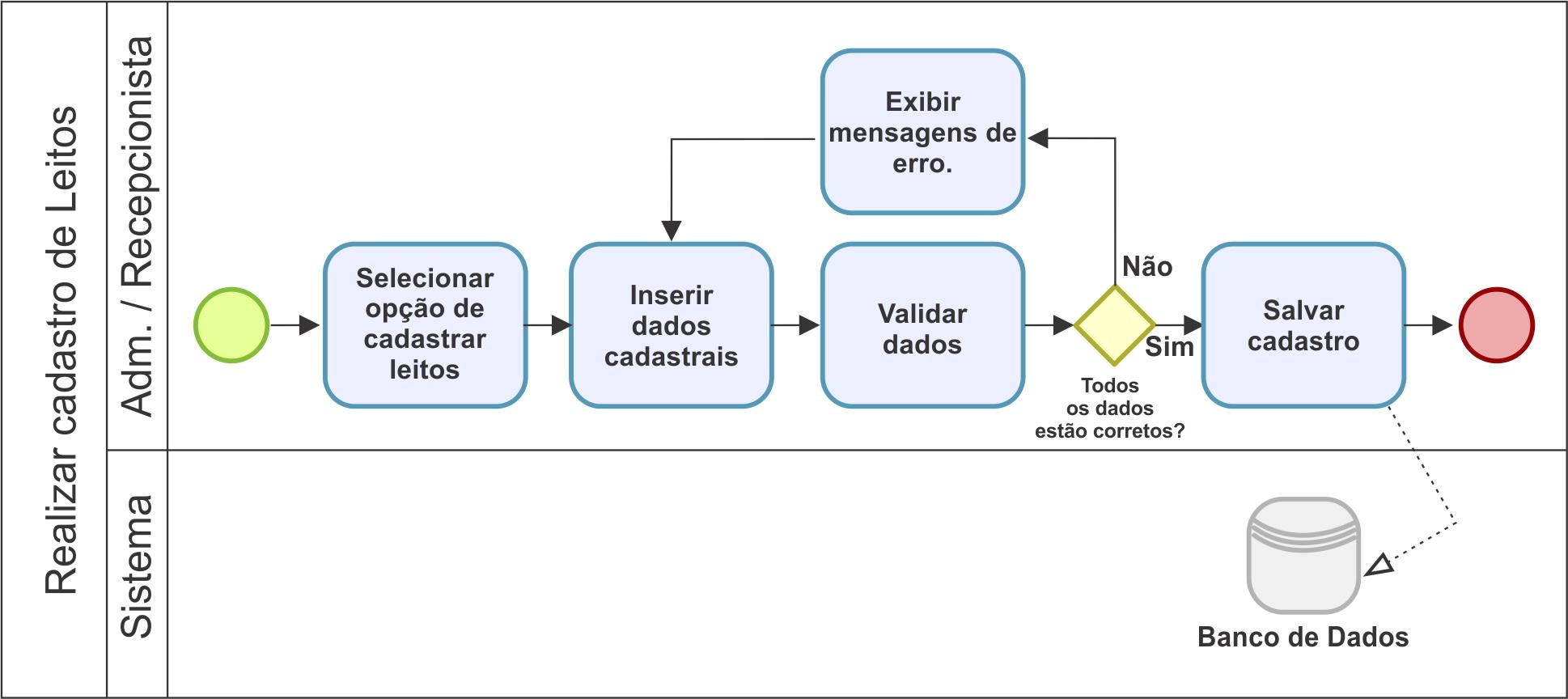
Esta seção tem como objetivo descrever os processos que foram modelados para a aplicação proposta. Vale ressaltar que cada processo representado implica em o usuário estar devidamente *logado* no sistema.

A Figura 2 exibe o processo denominado “Realizar cadastro da recepcionista”. O administrador, após adquirir o sistema, poderá realizar o seu cadastro no sistema, sendo este responsável por persistir os dados. A tarefa “Validar dados” tem como finalidade verificar se todos os campos obrigatórios foram digitados corretamente.

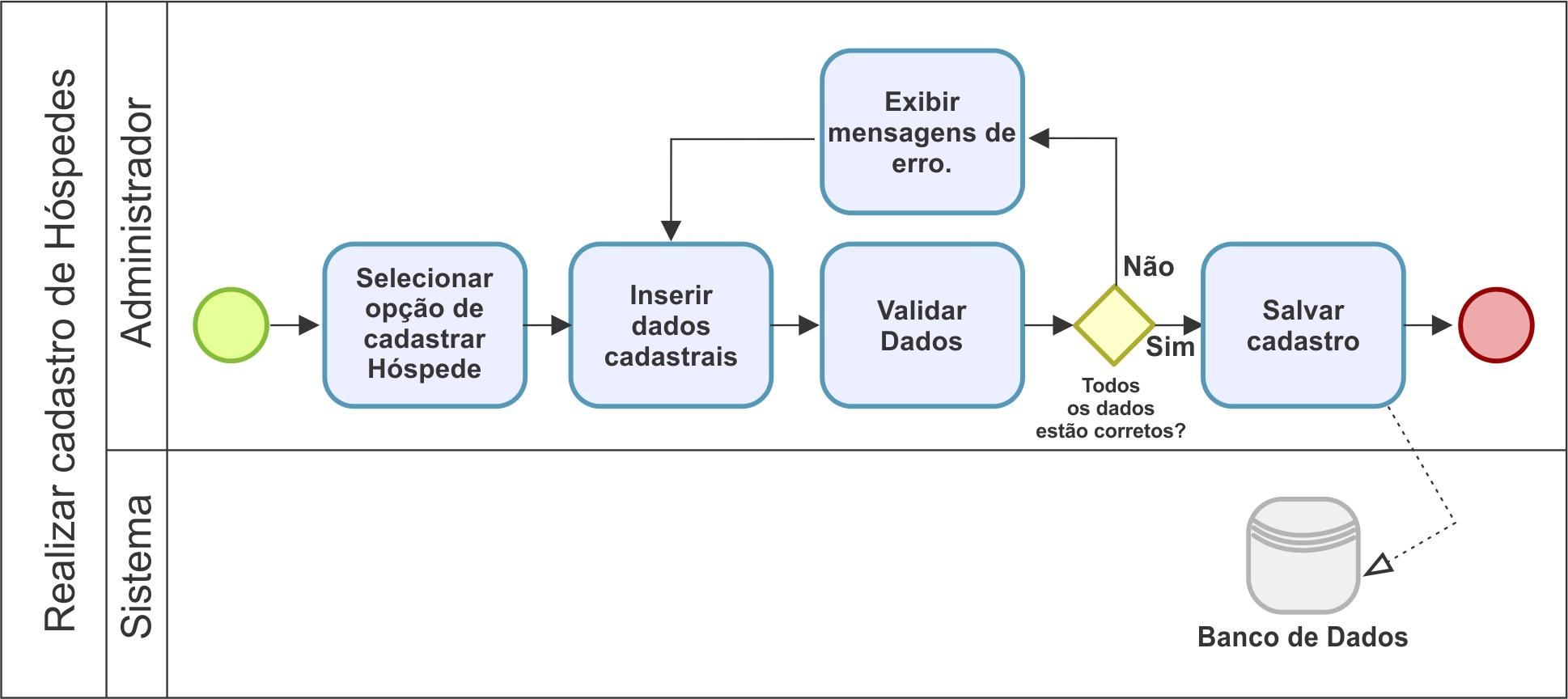
Em relação ao cadastro de leitos e de hóspedes no sistema, só poderá ser cadastrado se a recepcionista estiver cadastrada conforme a Figura 3 e 4.



*Figura 2 – Realizar cadastro de Recepcionista*

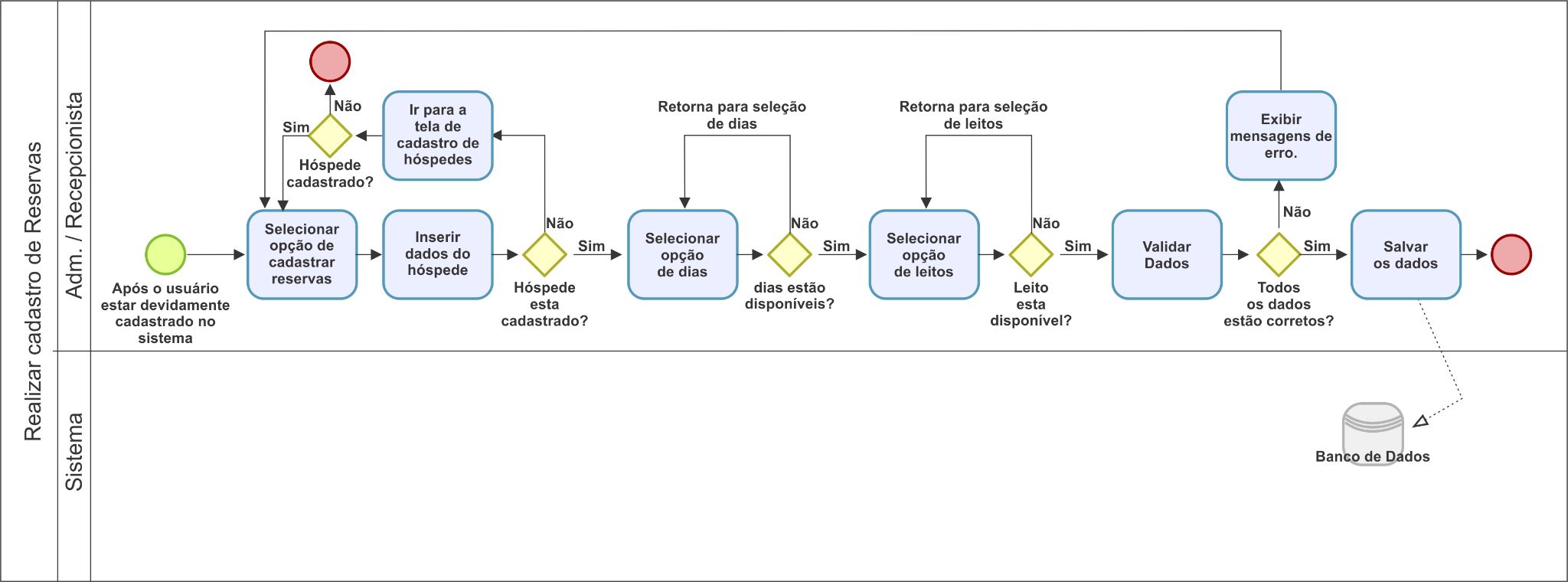


*Figura 3 – Realizar cadastro de leitos*



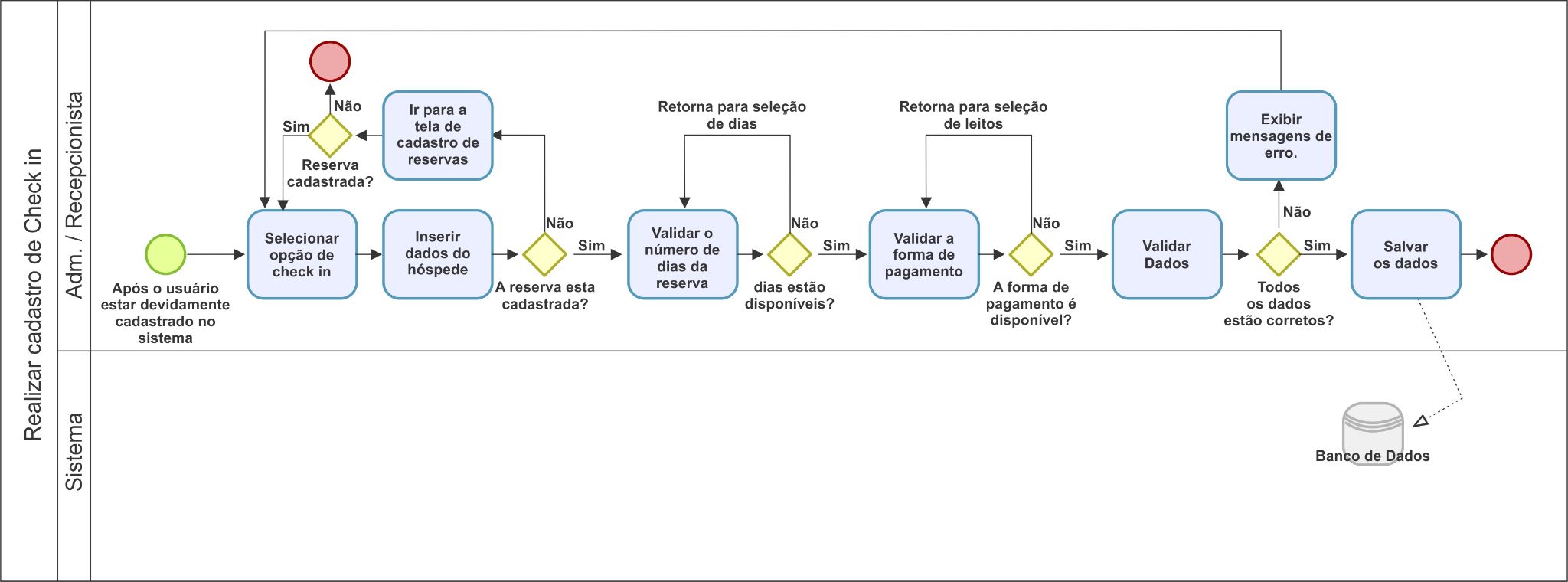
*Figura 4 – Realizar cadastro de hóspedes*

A Figura 5 representa a parte do sistema responsável pela reserva. Para a efetivação da reserva é necessário que o hóspede e os leitos estejam cadastrados e habilitados conforme representado na Figura.

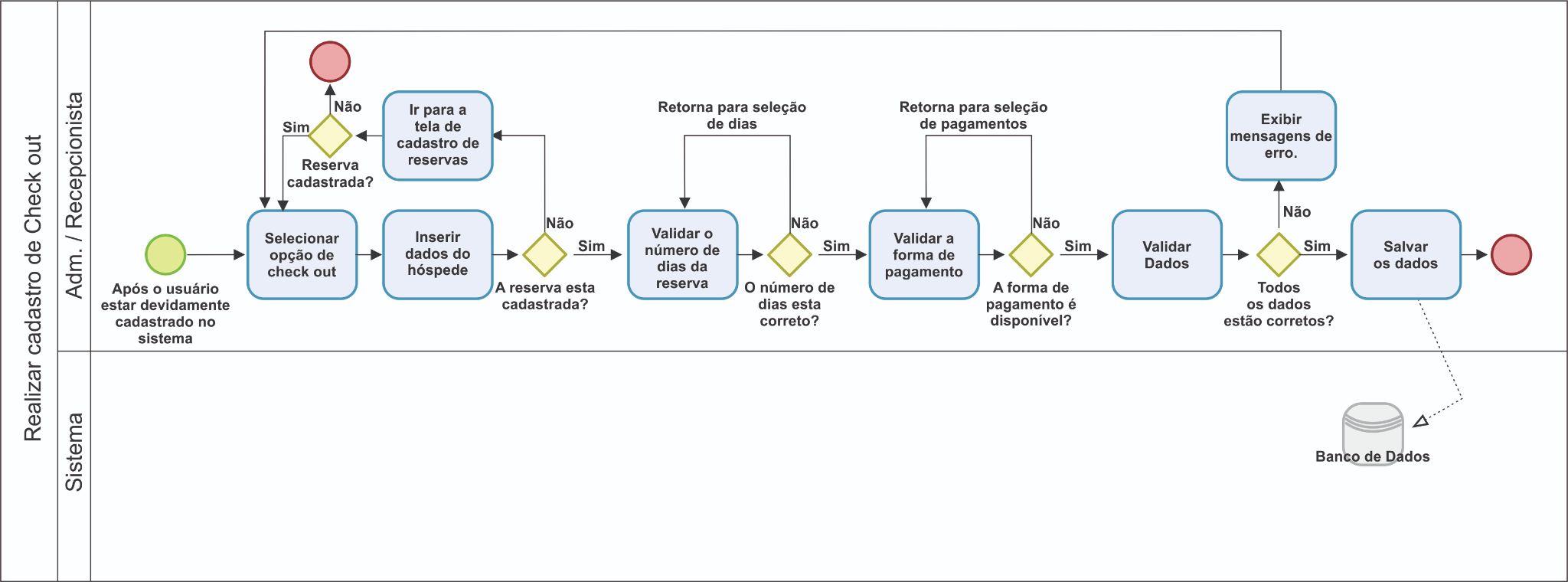


*Figura 5 – Realizar cadastro de reservas*

A Figura 6 representa a parte do sistema que efetiva a entrada do hóspede, o *check-in*. Para a efetivação do *check-in* é necessário que a reserva esteja no sistema. E a Figura 7 representa o *check-out*, onde será lançado os dias da reserva, as receitas daquela diária.



*Figura 6 – Realizar Check-in*

**

*Figura 7 – Realizar Check-out*

# Requisitos funcionais de sistema (RFS)

Requisitos funcionais descrevem as diversas funções que usuários e clientes querem ou precisam que o *software* forneça, ou seja, são requisitos ligados diretamente à funcionalidade que o sistema deve prover [2].

**OBS:** O acrônimo CRUD, listados em alguns requisitos funcionais de sistema, se refere às tarefas de *CREATE, READ, UPDATE e DELETE*, sendo respectivamente o processo de criar, ler, atualizar e deletar dados do sistema.

**[RF01] - CRUD HÓSPEDE**

O sistema possibilitará o cadastro de hóspedes contendo documento original com foto, endereço e telefone.

|  |  |
| --- | --- |
| **Atores** | Admin/ Recepcionista |
| **Entradas e pré-condições** | O usuário deverá inserir os seguintes dados obrigatórios: nome completo, CPF, data de nascimento, e-mail e telefone. |
| **Regras de negócio** | * O cadastro deve ser feito antes da reserva. * O CPF deve ser validado. * O e-mail deve ser validado. * O telefone não é um atributo obrigatório. |
| **Observações** | O processo de desativar um cliente no sistema será de forma lógica, entretanto todas as informações do mesmo devem permanecer no banco de dados. |
| **Prioridade** | Essencial |

**[RF02] - CRUD USUÁRIOS DO SISTEMA**

O sistema delimita acessos conforme o tipo de usuário: administrador ou recepcionista.

|  |  |
| --- | --- |
| **Atores** | Admin |
| **Entradas e pré-condições** | O administrador deverá inserir os seguintes dados obrigatórios: nome completo, endereço, CPF, data de nascimento, e-mail e telefone do usuário. |
| **Regras de negócio** | * Só o administrador pode cadastrar a recepcionista. * Tanto o administrador como a recepcionista podem cadastrar hóspedes. * A criação de usuário e senha deve ser enviado para o e-mail cadastrado pelo usuário. |
| **Observações** | O processo de desativar um usuário no sistema será de forma lógica, entretanto todas as informações do mesmo devem permanecer no banco de dados. |
| **Prioridade** | Essencial |

**[RF03] - CRUD LEITOS**

|  |  |
| --- | --- |
| **Atores** | Administrador e recepcionista |
| **Entradas e pré-condições** | Tanto o administrador como a recepcionista podem cadastrar uma reserva no sistema. |
| **Regras de negócio** | * Para cadastrar um leito é preciso ter um quarto disponível e vago. * O leito precisa ter um status: ativado, desativado, livre ou reservado. |
| **Observações** |  |
| **Prioridade** | Essencial |

**[RF04] - CRUD QUARTOS**

|  |  |
| --- | --- |
| **Atores** | Admin/Recepcionista |
| **Entradas e pré-condições** | O usuário deverá inserir o número de leitos no quarto, quais os tipos de leitos e qual a categoria do quarto: feminino, masculino ou misto. |
| **Regras de negócio** | * É obrigatório especificar a categoria (feminino, masculino ou misto). * O quarto tem três status: (cama vaga, completo e desativado). É obrigatório mostrar se tiver, o número de camas livres. |
| **Observações** | O processo de desativar um quarto no sistema será de forma lógica, entretanto todas as informações do mesmo devem permanecer no banco de dados. |
| **Prioridade** | Essencial |

**[RF05] - TELA LOGIN**

|  |  |
| --- | --- |
| **Atores** | Admin/Recepcionista |
| **Entradas e pré-condições** | O usuário deverá inserir seu login e senha para entrar no sistema. |
| **Regras de negócio** | * Todo usuário deve ser cadastrado pelo administrador para acessar o sistema. |
| **Observações** |  |
| **Prioridade** | Essencial |

**[RF06] - TELA HOME**

|  |  |
| --- | --- |
| **Atores** | Admin/Recepcionista |
| **Entradas e pré-condições** | O usuário deve estar logado no sistema. |
| **Regras de negócio** | * O sistema deve apresentar um menu lateral. * O tipo de usuário definirá as opções que aparecerão no menu. |
| **Observações** |  |
| **Prioridade** | Essencial |

**[RF07] - MAPA DE LEITOS**

|  |  |
| --- | --- |
| **Atores** | Administrador e recepcionista |
| **Entradas e pré-condições** | Tanto o administrador como a recepcionista podem acessar o mapa de leitos. |
| **Regras de negócio** | * O mapa de leitos deve mostrar todos os leitos, em quais quartos eles estão alocados e mostrar de cor diferente se está ocupado ou disponível. |
| **Observações** |  |
| **Prioridade** | Importante |

**[RF08] - CRUD RESERVAS**

|  |  |
| --- | --- |
| **Atores** | Administrador e recepcionista |
| **Entradas e pré-condições** | Tanto o administrador como a recepcionista podem cadastrar uma reserva no sistema. |
| **Regras de negócio** | * O leito tem que estar no sistema e disponível. * A reserva só é liberada mediante pagamento de 50% na reserva, desde que seja uma exigência do hostel. * O hóspede tem que estar cadastrado no sistema. * O *check-in* será iniciado no dia da entrada. * O *check-out* só finalizará a reserva quando for pago os outros 50% do valor. * O cancelamento da reserva só estorna o pagamento mediante ao aviso com mais de 24h, ou seja, em menos de 24h os 50% ficam para o hostel. |
| **Observações** | O processo de desmarcar uma reserva deve seguir as regras do hostel em questões burocráticas |
| **Prioridade** | Essencial |

**[RF09] - PAGAMENTOS**

|  |  |
| --- | --- |
| **Atores** | Administrador e recepcionista |
| **Entradas e pré-condições** | Tanto o administrador como a recepcionista podem lançar os pagamentos da estadia. |
| **Regras de negócio** | * Um pagamento só pode ser lançado no sistema se estiver relacionado a uma reserva. |
| **Observações** |  |
| **Prioridade** | Importante |

**[RF10] - FLUXO DE CAIXA**

|  |  |
| --- | --- |
| **Atores** | Administrador e recepcionista |
| **Entradas e pré-condições** | Tanto o administrador como a recepcionista podem lançar as receitas e as despesas. |
| **Regras de negócio** | * Uma receita só pode ser lançada no sistema se estiver relacionada a uma reserva. * Qualquer despesa pode ser lançada no sistema. * O sistema deve calcular um fluxo de caixa simples. |
| **Observações** |  |
| **Prioridade** | Importante |

**[RF11] - GERAR RELATÓRIO DE OCUPAÇÃO**

|  |  |
| --- | --- |
| **Atores** | Administrador |
| **Entradas e pré-condições** | Apenas o administrador tem acesso aos relatórios. |
| **Regras de negócio** | * Os relatórios de ocupação deverão mostrar quantas reservas foram solicitadas, quantas foram realizadas e quantas foram canceladas. * Os relatórios poderão ser consultados por default no dia vigente, mas poderão ser consultados dentro da semana, no mês, no ano e no intervalo que o administrador desejar. |
| **Observações** |  |
| **Prioridade** | Importante |

**[RF12] - GERAR RELATÓRIO DE FINANCEIRO**

|  |  |
| --- | --- |
| **Atores** | Administrador |
| **Entradas e pré-condições** | Apenas o administrador tem acesso aos relatórios. |
| **Regras de negócio** | * Os relatórios de ocupação deverão mostrar a partir das reservas solicitadas, realizadas e canceladas o fluxo do caixa, quanto de receita foi gerado. * Os relatórios poderão ser consultados por default no dia vigente, mas poderão ser consultados dentro da semana, no mês, no ano e no intervalo que o administrador desejar. |
| **Observações** |  |
| **Prioridade** | Importante |

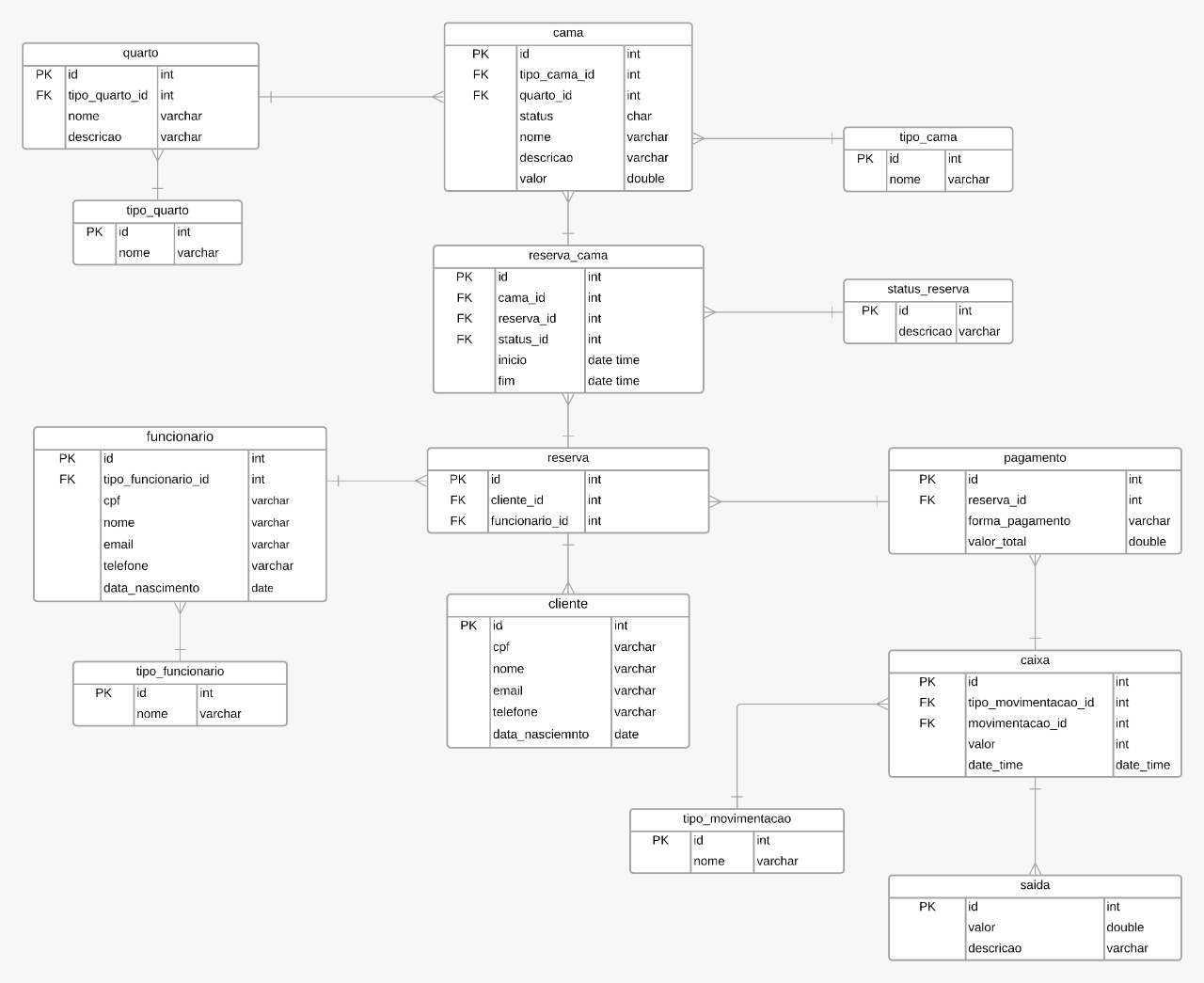
# Requisitos Não Funcionais (RNF)

Um Requisito Não Funcional é uma não funcionalidade do sistema, mas que precisa ser realizado para que o software atenda seu propósito. Esse tipo de requisito define como o sistema fará. A seguir serão apresentados os RNF do **Sistema de Gerenciamento de Hostel**.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | NOME | DESCRIÇÃO | E | I | D |
|  | Framework Front-end | O Framework a ser utilizado no desenvolvimento do Front-end do sistema será o NextJS; | **X** |  |  |
|  | Sistema de gerenciamento de banco de dados (SGBD) | O SGBD a ser utilizado será o PostgreSQL; | **X** |  |  |
|  | Framework Back-end | O Framework a ser utilizado no desenvolvimento do Back-end do sistema será o Django; | **X** |  |  |
|  | Autenticação | A fim de garantir a segurança do sistema, a autenticação será através de login e senha. | **X** |  |  |
|  | Funcionamento online | O sistema funcionará somente se tiver acesso à internet | **X** |  |  |
| [RFN 06] | Linguagem de programação | Serão utilizadas as seguintes linguagem de programação e marcação: Python, TypeScript, HTML, CSS | **X** |  |  |

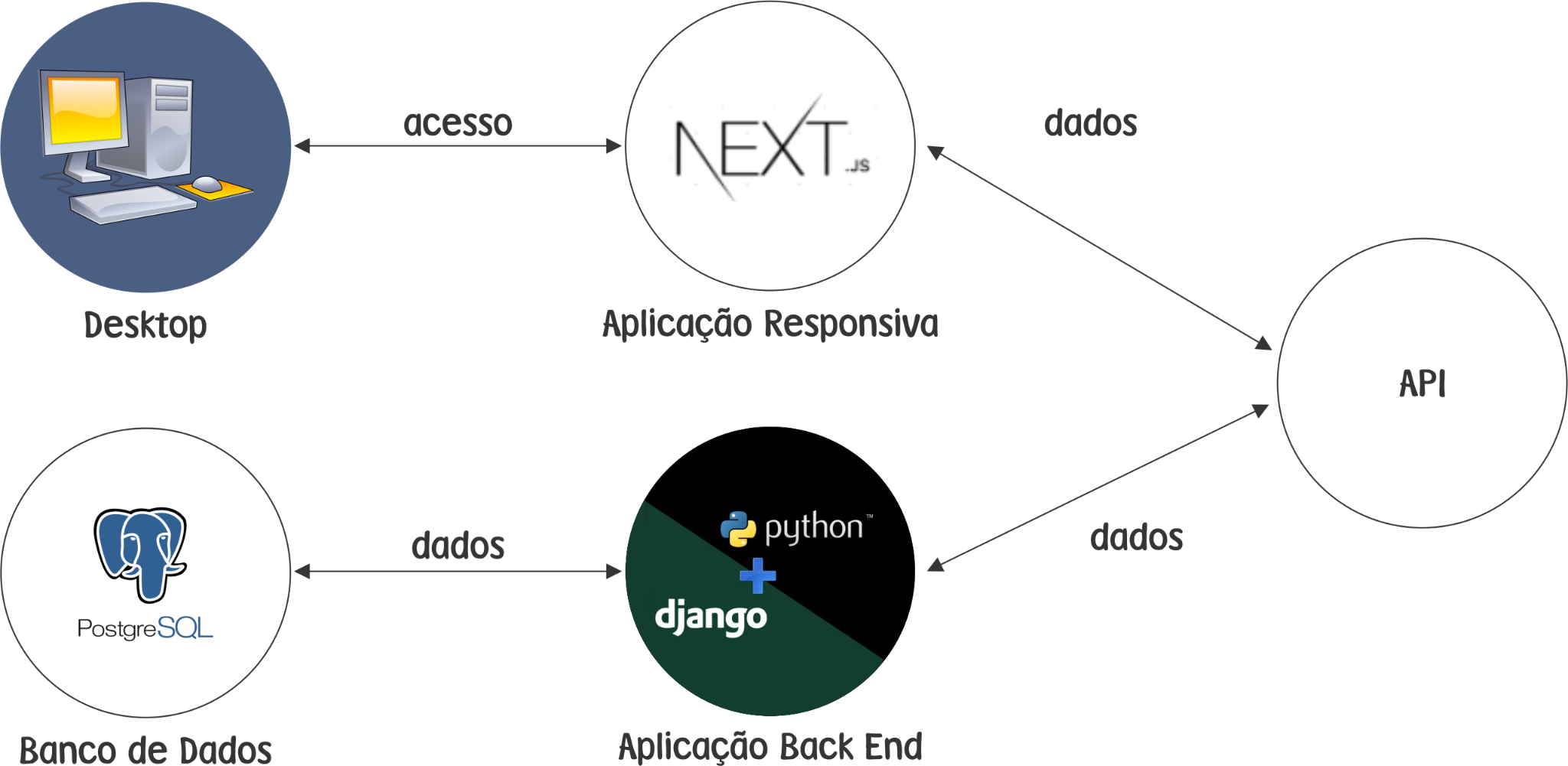
# Modelo Conceitual do Banco de Dados

O modelo conceitual do banco de dados para o **Sistema de Gerenciamento de Hostel** é apresentado na Figura 13.



# Arquitetura da Aplicação

O **Sistema de Gerenciamento de Hostel** terá sua parte Front-end desenvolvida utilizando o framework Next.Js e utilizará diversos módulos do serviço Phyton e Django para implementar as funções do Back-end. Uma exemplificação em alto nível é exibida na Figura .



*Figura – Arquitetura Simplificada*