****

|  |
| --- |
| Engenharia de Software 2016/2017 |
| Especificação do Sistema – alto nível |

Turma: BrightStart

Grupo nº 1

|  |  |
| --- | --- |
| **Nº** | **Nome:** |
| 180288023 | André Castro |
| 180288028 | André Batista |
| 180288041 | João Daniel |
| 180288036 | Luís Fernandes |
| 180288033 | Tiago Henriques |

**Versões do Trabalho**

|  |  |  |  |
| --- | --- | --- | --- |
| Versão | Data | Autor | Descrição |
| 1 | 12-04-2019 | André Castro | Entregavel até a data |
| 2 |  |  |  |
|  |  |  |  |
|  |  |  |  |

[1 SUMARIO EXECUTIVO 4](#_Toc5991819)

[2 INTRODUÇÃO 5](#_Toc5991820)

[3 DESENHO DE ALTO NÍVEL 6](#_Toc5991821)

[3.1 ARQUITETURA GERAL 6](#_Toc5991822)

[3.2 ARQUITETURA LÓGICA 6](#_Toc5991823)

[3.3 DIAGRAMA DE CLASSES DE DESENHO 7](#_Toc5991824)

[3.4 INTERFACE COM O UTILIZADOR 8](#_Toc5991825)

[3.4.1 Introdução 8](#_Toc5991826)

[3.4.2 Mockups 9](#_Toc5991827)

[3.4.3 Normas 12](#_Toc5991828)

[3.4.4 Diagrama Geral de Navegação 12](#_Toc5991829)

[3.4.5 Matriz de acessos 12](#_Toc5991830)

[3.5 PERSISTÊNCIA 13](#_Toc5991831)

[3.5.1 Introdução 13](#_Toc5991832)

[3.5.2 Modelo Relacional 14](#_Toc5991833)

[3.6 ARQUITETURA FÍSICA 15](#_Toc5991834)

[3.6.1 Introdução 15](#_Toc5991835)

[3.6.2 Diagrama de Componentes 15](#_Toc5991836)

[3.6.3 Diagrama de Instalação 16](#_Toc5991837)

[3.7 NORMAS DE CODIFICAÇÃO DA APLICAÇÃO 16](#_Toc5991838)

# SUMARIO EXECUTIVO

ESTech, é a aplicação que tem como objetivo vir a revolucionar por completo o sistema de segurança, requisições de materiais, chaves, salas, e sistema de relatórios e análise estatística presentes na EST-IPS, que, de momento, é tudo feito por papel e caneta. Com esta aplicação, este processo será muito mais eficiente, sendo que a gestão de salas disponíveis vai ser mais intuitiva, o acesso a estatísticas diversas mais completo e user-friendly, e o sistema de requisição de materiais mais fácil de usar e com históricos mais pertinentes, tendo mais informação relevante.

Este projeto está a ser desenvolvido por 5 elementos da turma da 1º Edição do Programa BrightStart da IPS-EST, que, bem como o resto da turma, se questionou muito sobre o sistema implementado (ou a falta dele) na escola.

No futuro temos o objetivo de vir a implementar esta aplicação no resto das escolas do agrupamento do IPS, de forma a modernizar a forma arcaica de como este tipo de processos é gerido.

Claro que todo este projeto não será de fácil implementação, dado o tempo reduzido que temos à nossa disposição e com a forma de como as pessoas envolvidas no sistema atual hão de lidar com este grande impacto tecnológico no seu dia-a-dia.

O desenvolvimento da aplicação será feito em conjunto entre vários ambientes, sendo a base de dados proporcionada em MongoDB, que fará com que o desempenho do sistema não seja de grande preocupação.

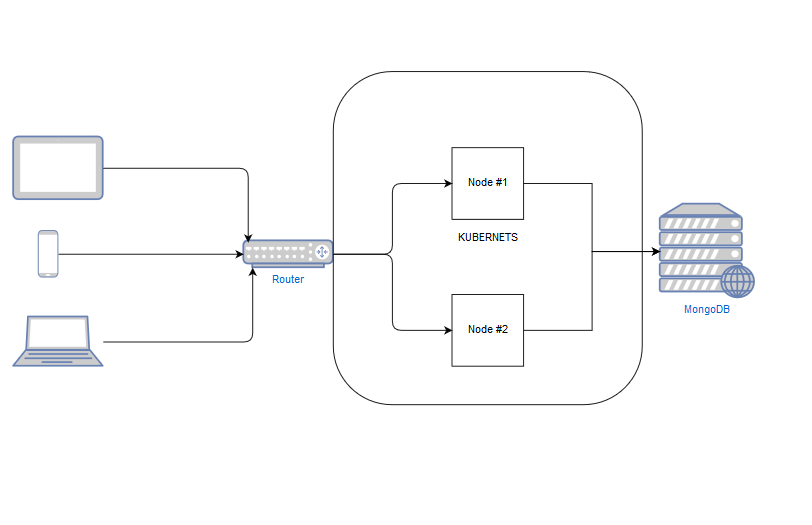
Acreditamos muito no desenvolvimento tecnológico deste tipo de entidades como o IPS, que por vezes parecem estar presas no tempo com os seus sistemas completamente out-dated para as necessidades da atualidade.

# INTRODUÇÃO

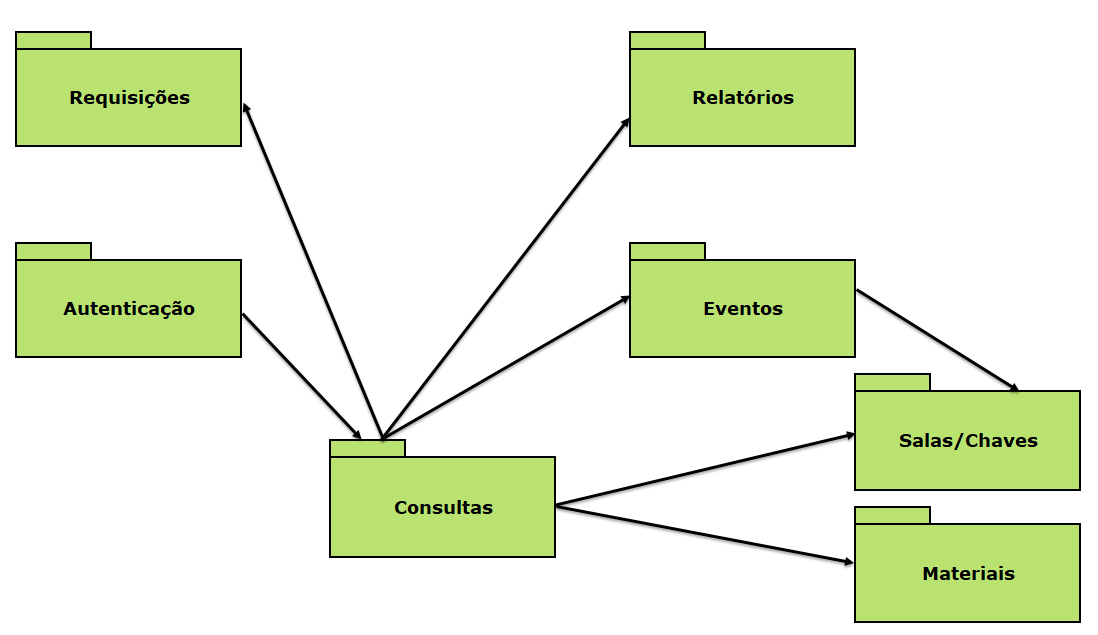
Este documento tem como objetivo dar um overview do funcionamento e lógica por de trás do projeto. Sendo isto um overview, o nível de detalhe é de alto nível de forma a que seja mais acessível a um nível funcional.

# DESENHO DE ALTO NÍVEL

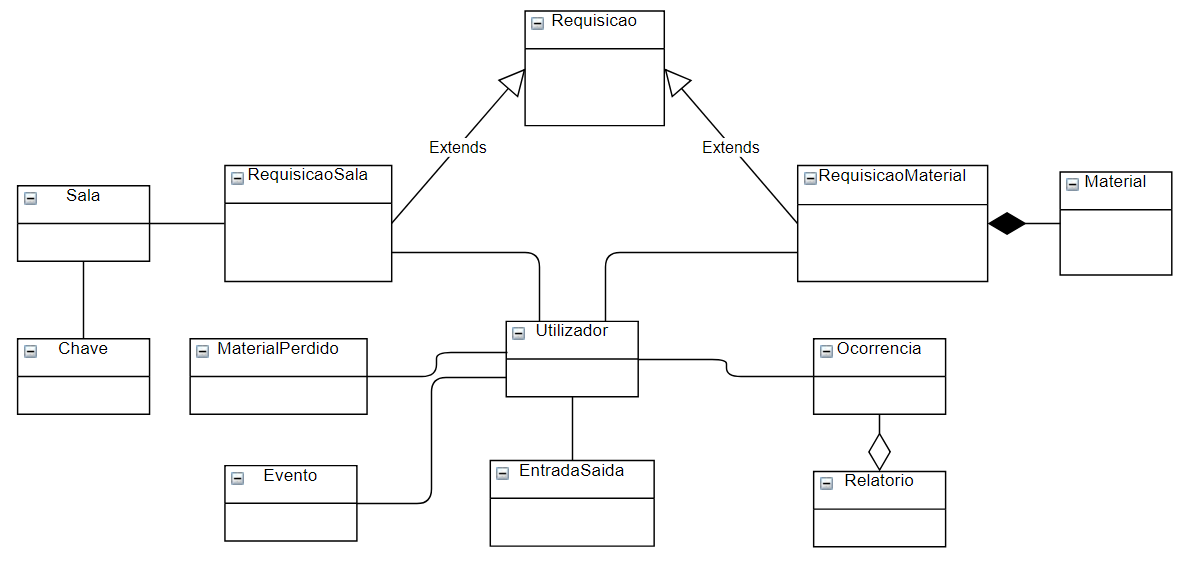
## ARQUITETURA GERAL



## ARQUITETURA LÓGICA



## DIAGRAMA DE CLASSES DE DESENHO



## INTERFACE COM O UTILIZADOR

### Introdução

O nosso web design vai ser constituído por uma estrutura simples, com uma boa e adaptada visibilidade para ser de fácil acesso tanto a partir

do computador como do telemóvel ou do tablet.

Uma boa visibilidade trata-se por todos os ícones serem consideravelmente grandes e fáceis de manusear a partir do telemóvel, etc.

Esta aplicação estará apenas disponível para alunos do ips, qualquer pessoa que não tenha acesso ao cartão de aluno, não pode fazer login.

Esta autenticação irá ter duas formas de ser executada:

-Manualmente: Inserir credenciais para entrar na conta

-Com o cartão: Passar o cartão num leitor de cartões e aceder automaticamente à sua conta

O formato do nosso website vai ser one page, apenas uma página com todas as funcionalidades disponíveis sem ter que aceder a outras.

A página irá ter vários tópicos numa barra de menu ao entrar no site no canto superior.

Cada tópico irá estar constituído por anchors, a partir desta funcionalidade basta um click num dos tópicos, que será transportado

para o menu desejado.

### Mockups

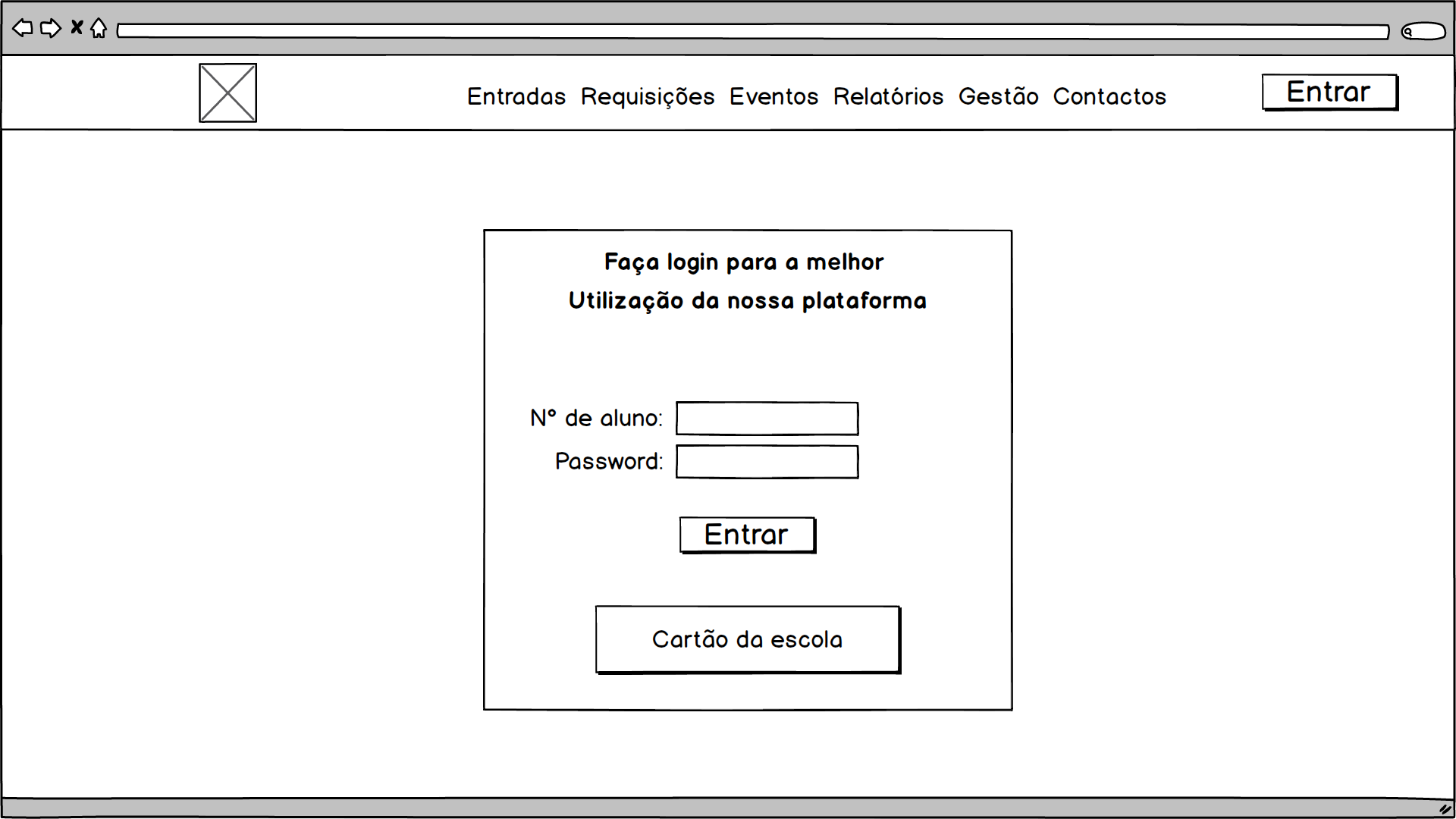


Figura 1 - Página home

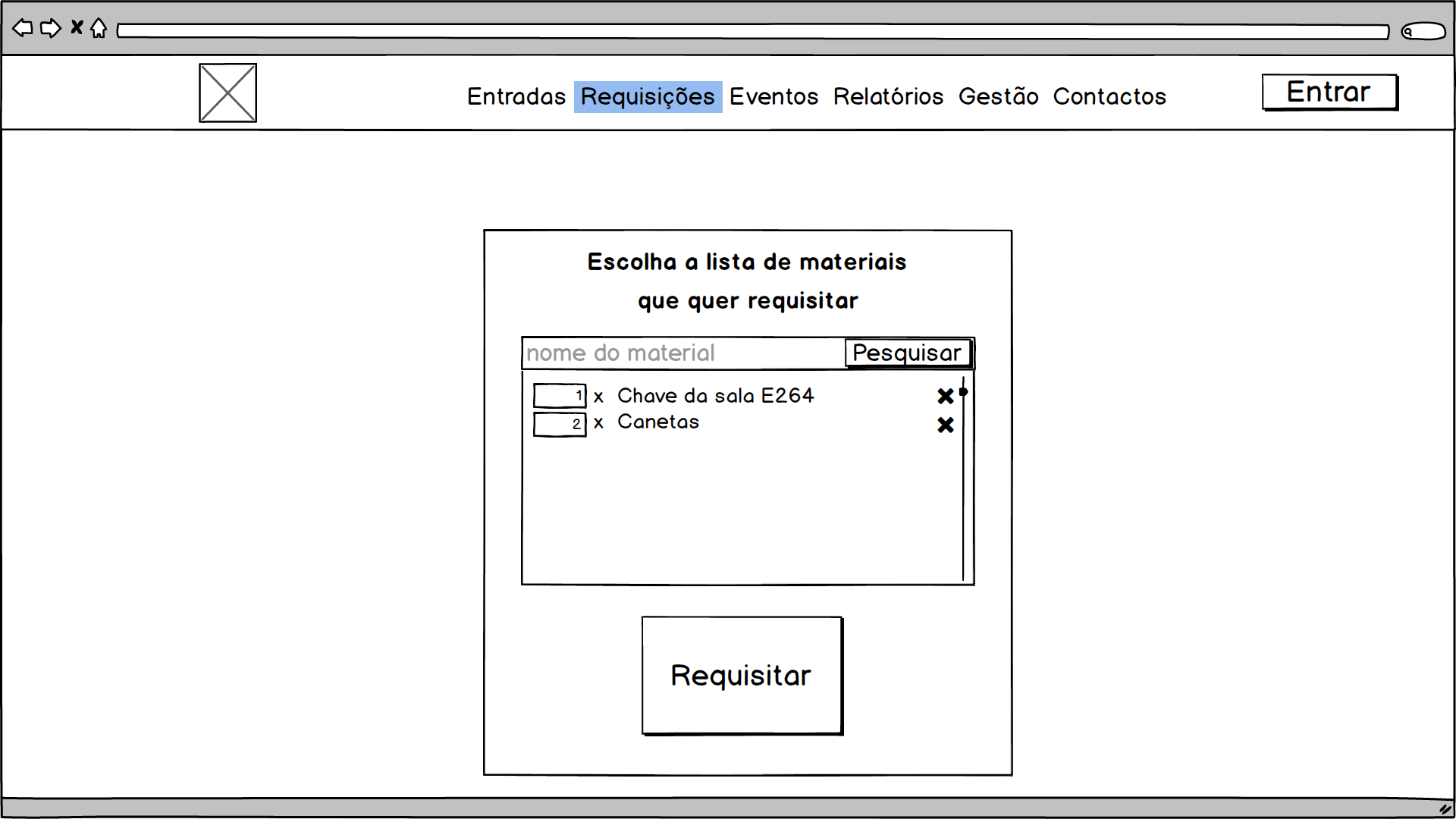


Figura 2 - Pagina de requisitar

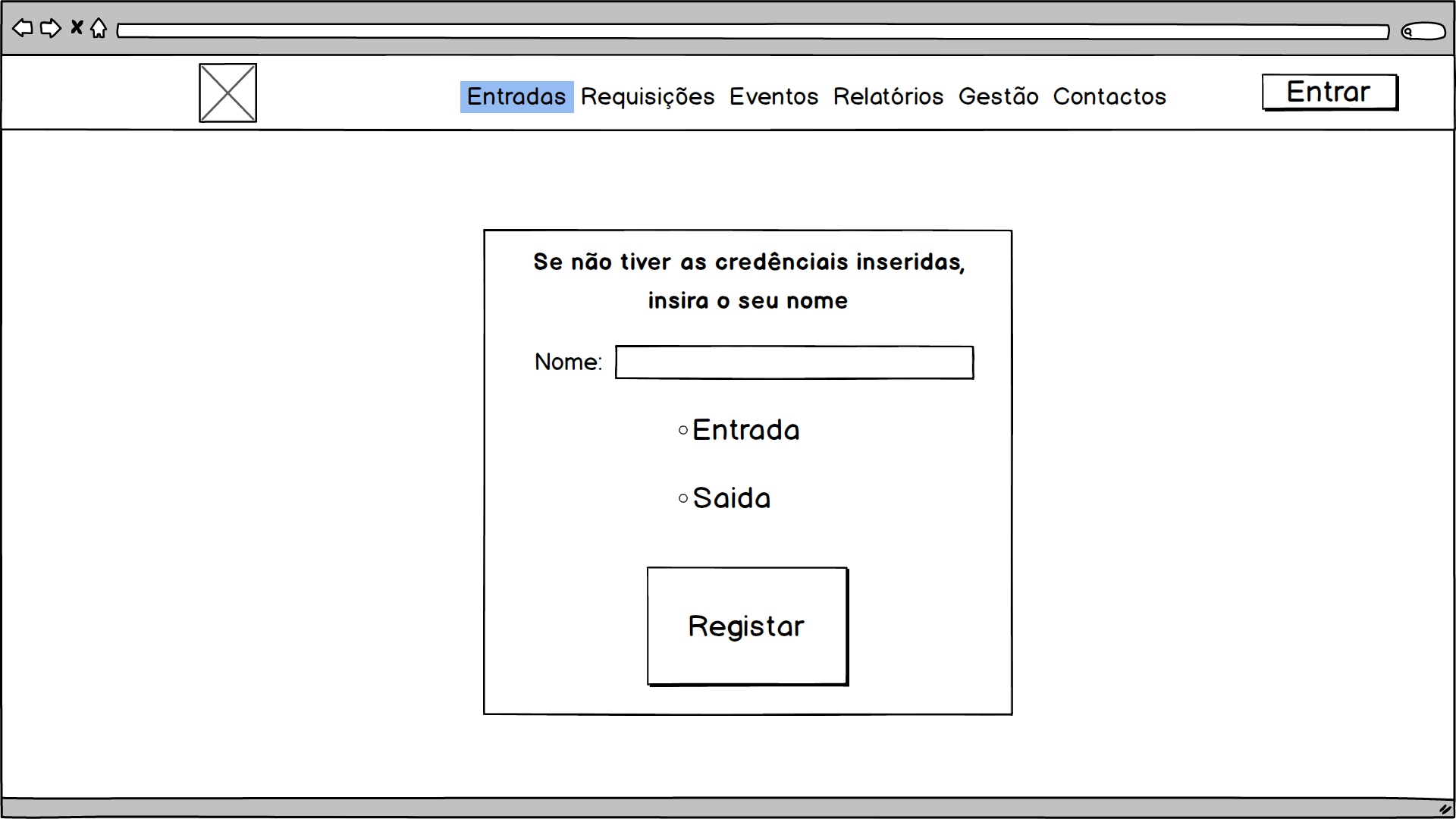


Figura 3 - Página Entradas

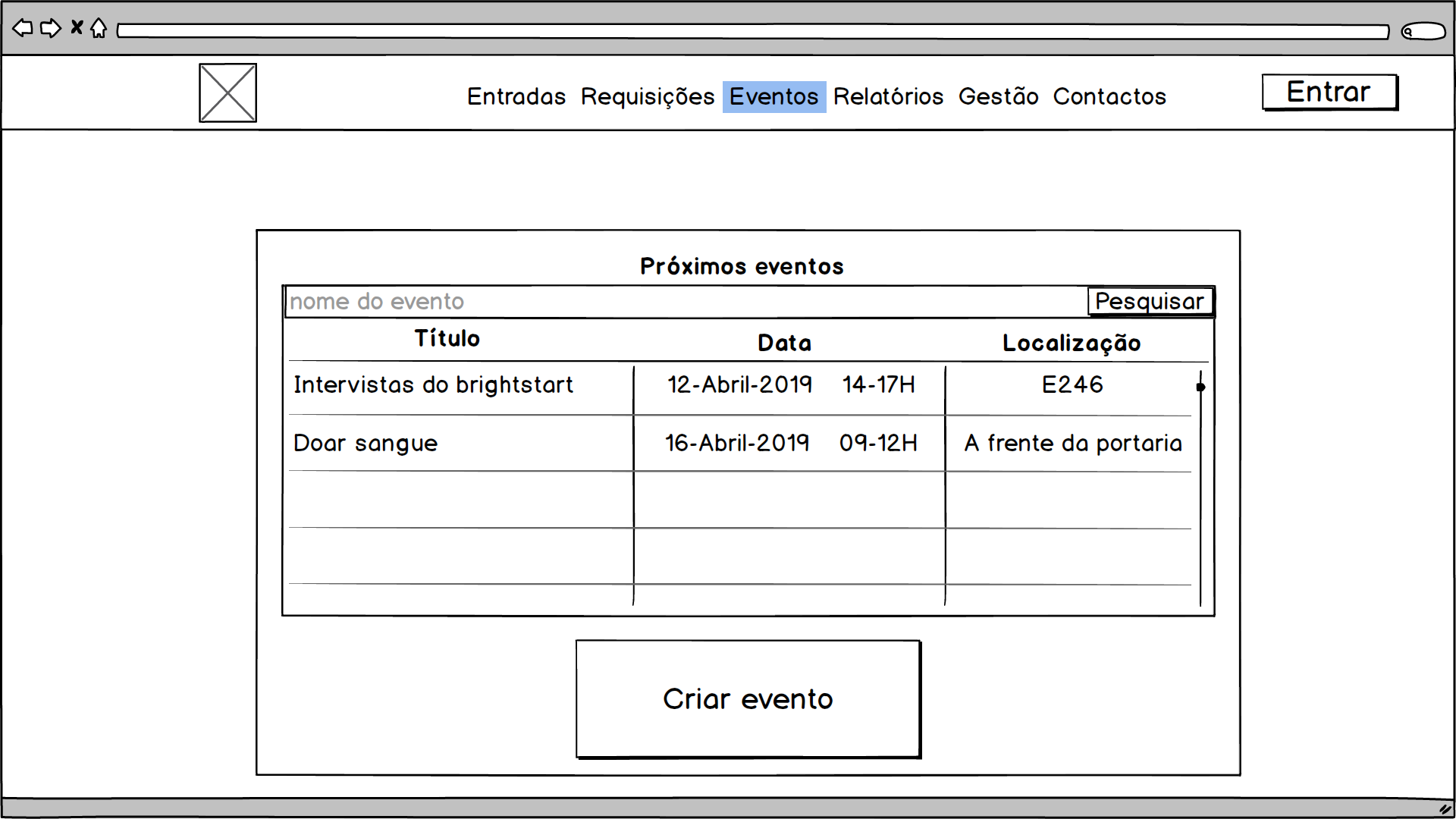


Figura 4 - Página de Eventos

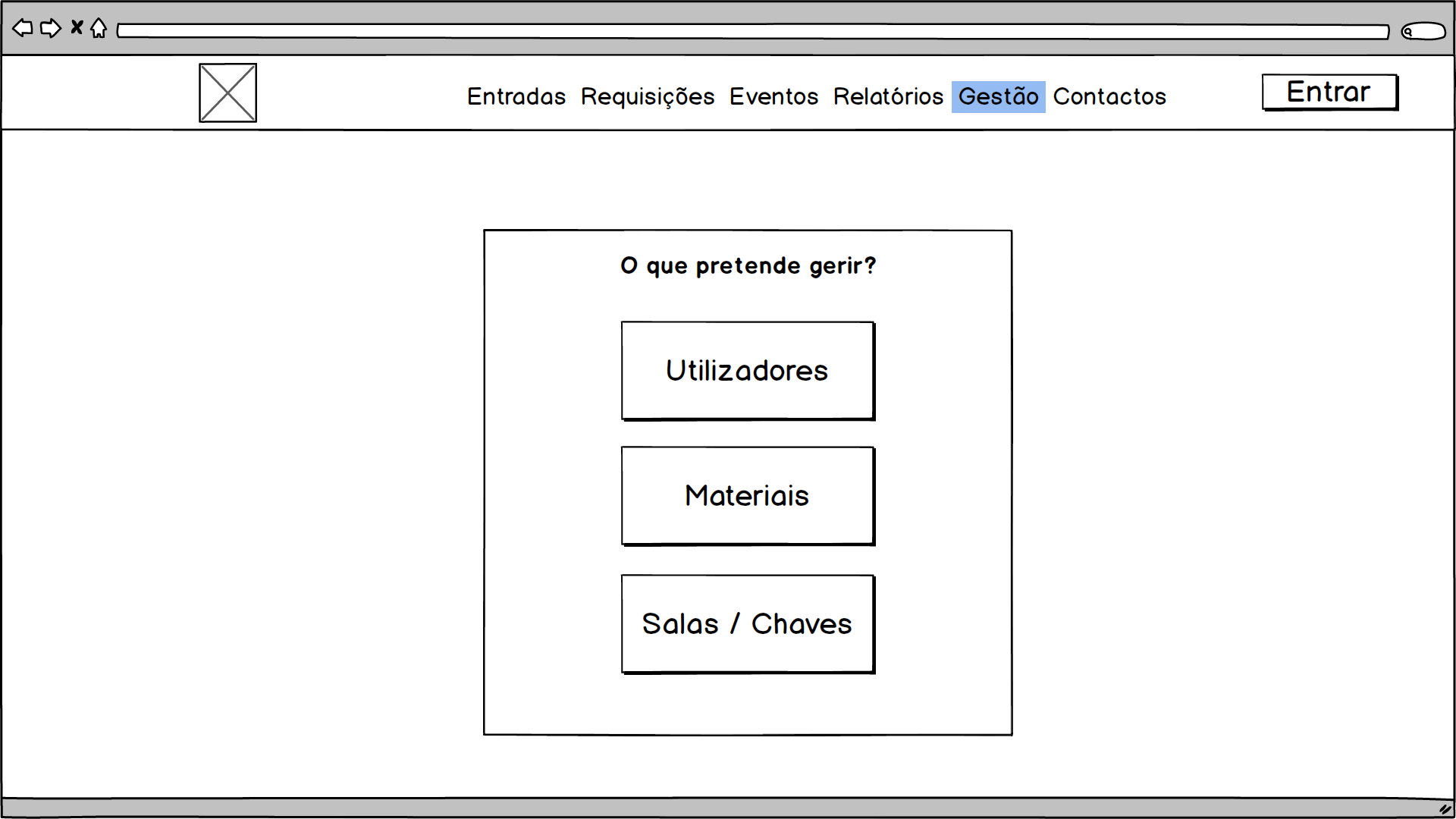


Figura 5 - Página de gestão

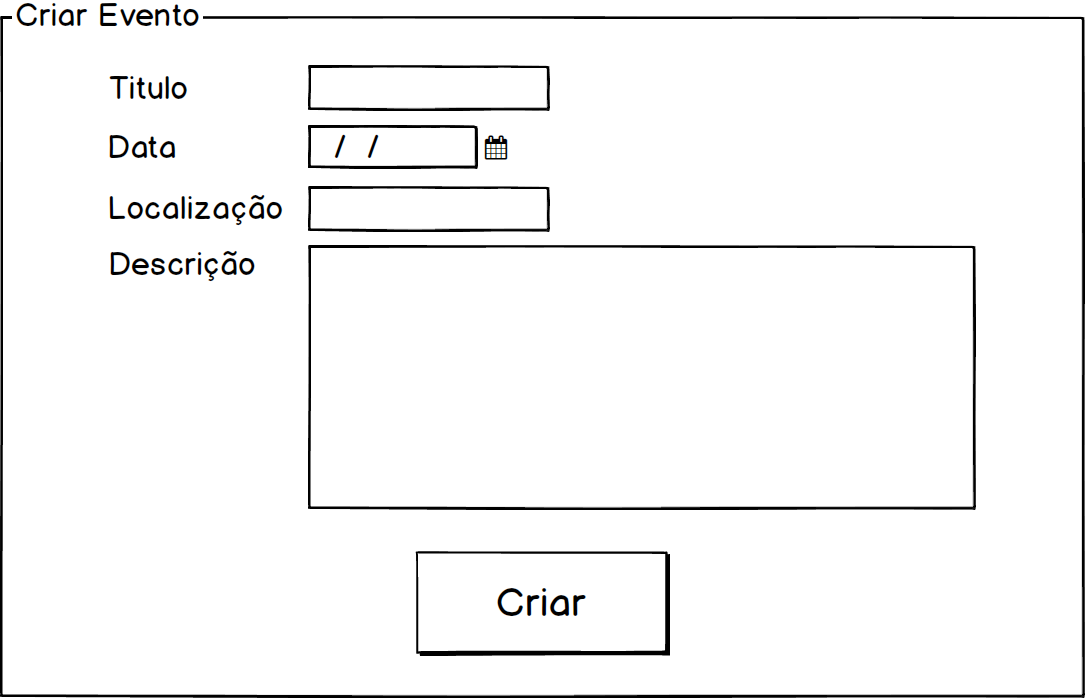


Figura 6 - InnerPopup de criar evento



Figura 7 - Página de Relatórios

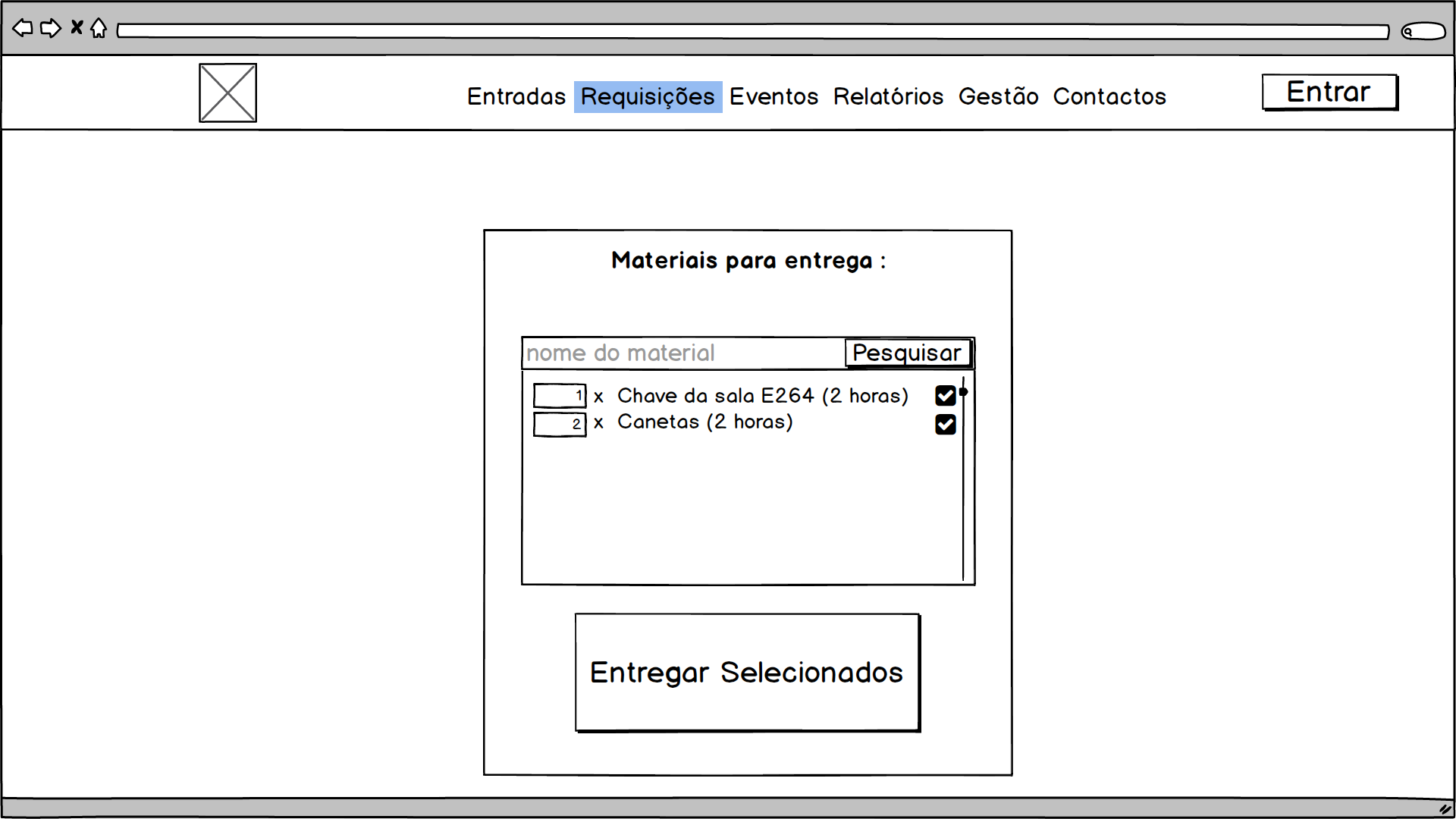


Figura 8 - Página de entregas

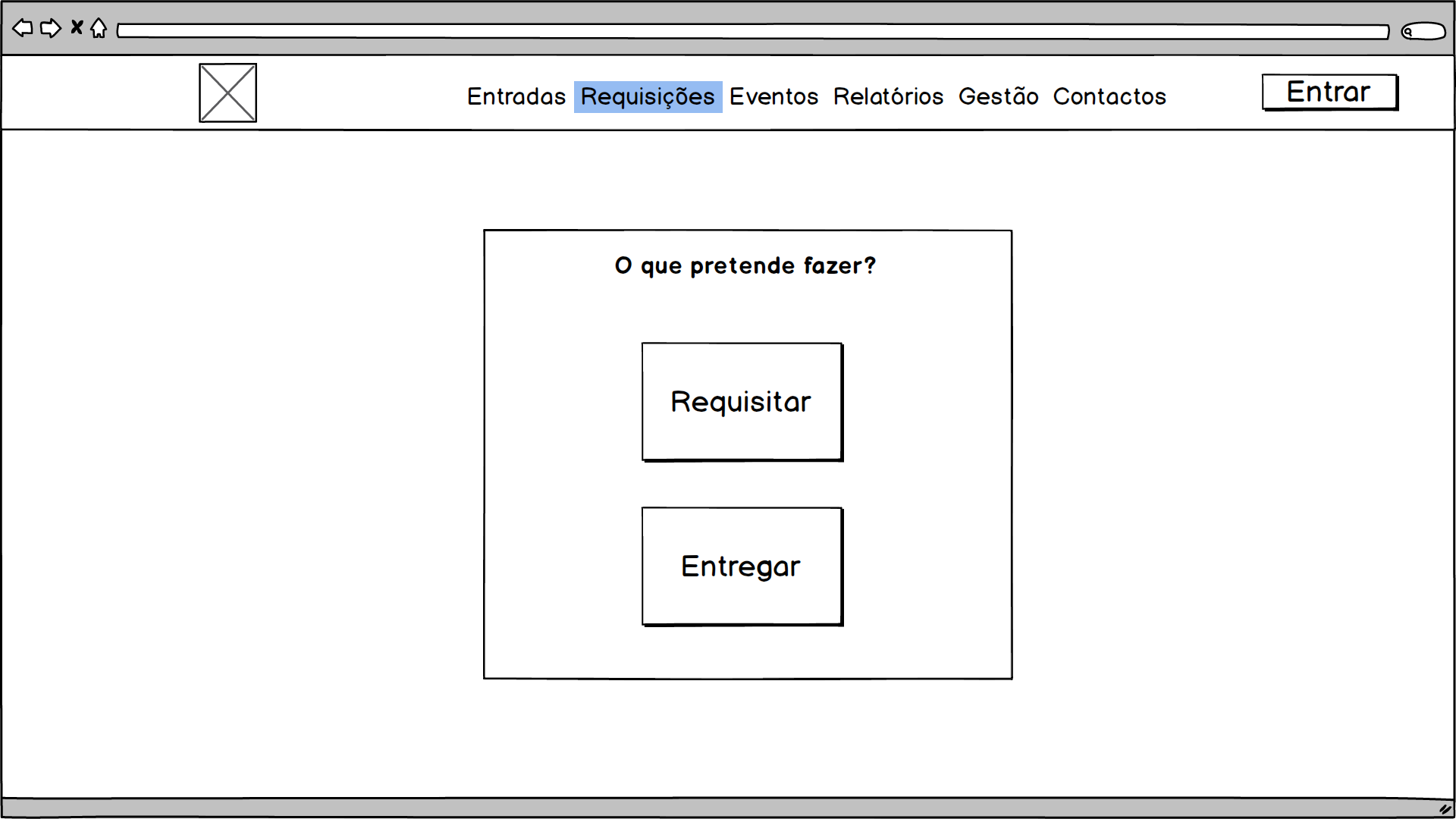


Figura 9 - Página de Menu de requisições

### Normas

Fonte: Balsamiq sans

Tamanho da fonte: entre 28 até 40 na maioria dos casos

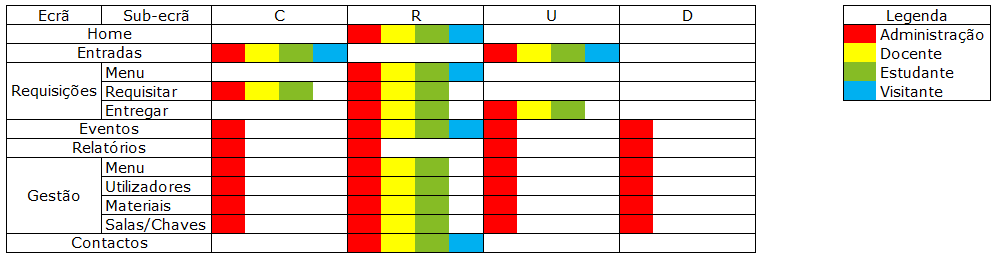
Palete de cores :

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |

Menu no topo com o logotipo à esquerda.

### Diagrama Geral de Navegação

### Matriz de acessos



## PERSISTÊNCIA

### Introdução

Definição

O MongoDB é um sistema de gestão de base de dados (SGBD),é open source e usa um modelo de base de dados orientado a documentos que suporta várias formas de dados.

É uma das inúmeras tecnologias de base de dados não relacionais que surgiram em meados da década de 2000 sob o NoSQL banner para uso em de big data aplications e outros trabalhos de processamento envolvendo dados que não se encaixam bem em um modelo relacional rígido. Em vez de usar tabelas e linhas como em bases de dados relacionais, a arquitetura do MongoDB é composta de coleções e documentos.



Como funciona

Um registo no MongoDB é um documento, que é uma estrutura de dados composta de pares de campos e valores.

Os documentos MongoDB são semelhantes aos objetos Notação de objeto JavaScript, mas usam uma variante chamada JSON binário (BSON) que acomoda mais tipos de dados.

Os campos nos documentos são semelhantes às colunas em uma base de dados relacional, e os valores que eles contêm podem ser uma variedade de tipos de dados, incluindo outros documentos, matrizes e matrizes de documentos, de acordo com o manual de utilizador do MongoDB.

Os documentos, que também devem incorporar uma chave primária como um identificador exclusivo, são a unidade básica de dados no MongoDB. As coleções contêm conjuntos de documentos e funcionam como o equivalente de tabelas de basas de dados relacionais.

As coleções podem conter qualquer tipo de dados, mas a restrição é que os dados em uma coleção não podem ser distribuídos em bases de dados diferentes.

O shell mongo é uma interface JavaScript interativa com o MongoDB, que permite aos usuários consultar e atualizar dados e realizar operações administrativas.

O shell é um componente padrão das distribuições de código aberto do MongoDB. Uma vez instalado o MongoDB, os usuários conectam o shell mongo às instâncias do MongoDB em execução.

O formato de armazenamento de dados e intercâmbio de dados BSON usado no MongoDB fornece uma representação binária de documentos semelhantes a JSON. O sharding automático é outro recurso importante que permite que os dados em uma coleção MongoDB sejam distribuídos em vários sistemas para escalabilidade horizontal, conforme os volumes de dados e os requisitos de taxa de transferência aumentam.

O NoSQL DBMS usa uma arquitetura mestre única para consistência de dados, com bancos de dados secundários que mantêm cópias do banco de dados primário. As operações são automaticamente replicadas para esses bancos de dados secundários para failover automático.

### Modelo Relacional

****

## ARQUITETURA FÍSICA

### Introdução

Neste projeto será o Kubernetes,

O Kubernetes ajuda a melhorar a eficiência e diminuir os recursos necessários nas equipas de desenvolvimento

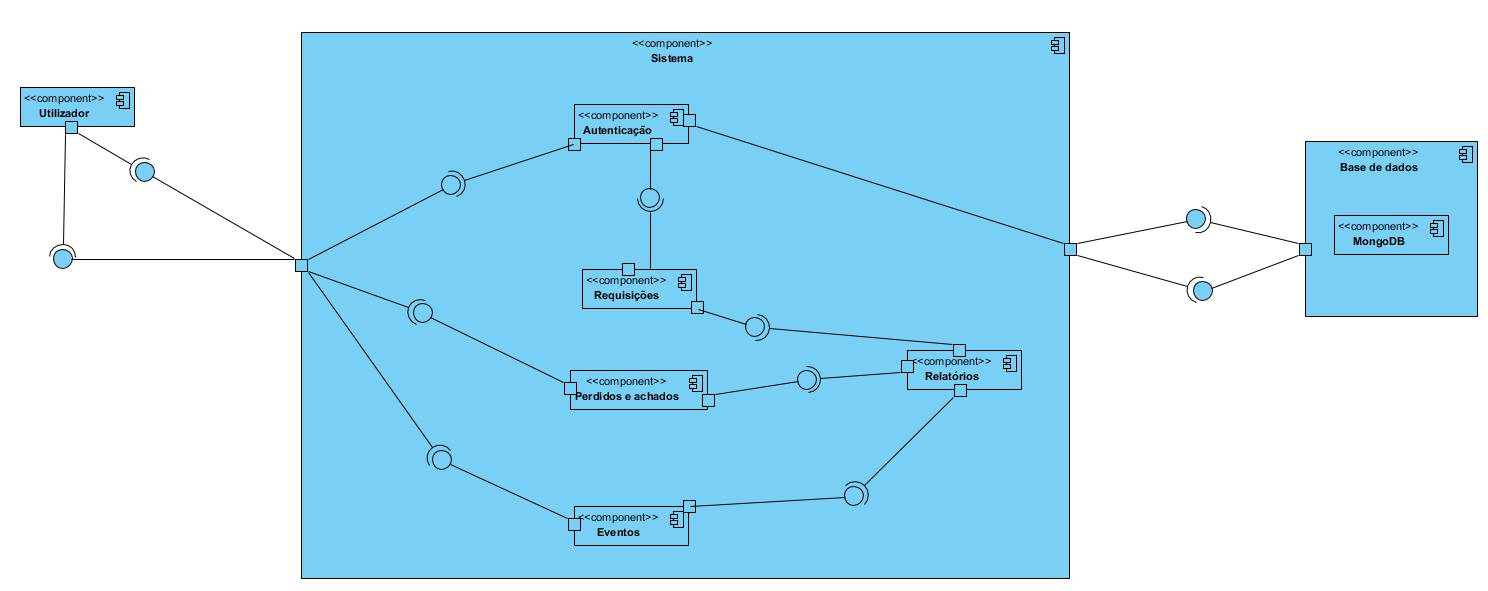
Para entender Kubernetes, temos que entender containers.

Os containers permitem que processos rodem de forma isolada num host com o mesmo S.O (Sistema Operativo)

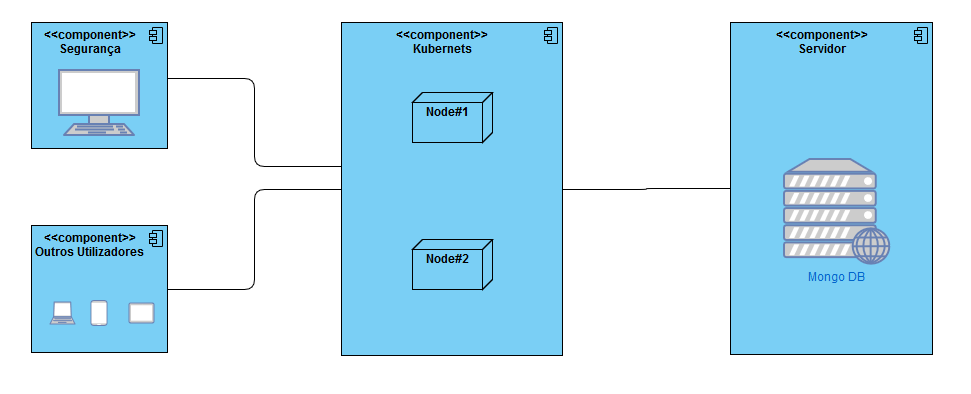
Outro ponto positivo dos containers é que numa aplicação com varias funcionalidades independentes, é possível que partes do software sejam geridas por equipas diferentes ou até mesmo com feito com linguagens diferentes

Outro Ponto importante é o conceito de cluster que não mais é do que um conjunto de maquinas por meio do qual os containers são executados

### Diagrama de Componentes



### Diagrama de Instalação



## NORMAS DE CODIFICAÇÃO DA APLICAÇÃO

Para normas de codificação vamos utilizar as normas de programação da especificação do ECMAScript 2018:

Para começar vamos utilizar:

* A sintaxe da linguagem JavaScript (regras de analise, palavras-chave, controlo do fluxo entre outros)
* Mecanismos para tratamento de erros(throw, try-catch, e criação de tipos de erros criados pelo programador)
* Tipos de Dados (boolean ,number ,string ,function ,object entre outros)
* O objeto global ( window )
* Mecanismos de herança prototype-based
* Objetos e funções OTB ( out of the box) (JSON, Math, Array.prototype, Object introspection
* Strict Mode