

**PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS**  
**Graduação em Sistemas de Informação**

**TECNOLOGIAS WEB:**  
**EVENTOS\_ACLB**

**Autor:       Davi Felipe Silva de Oliveira**

**08 de novembro de 2017**  
**Belo Horizonte**

## SUMÁRIO

<b>1.</b>	<b>Apresentação .....</b>	<b>3</b>
<b>1.1.</b>	<b>Problema .....</b>	<b>3</b>
<b>1.2.</b>	<b>Objetivos do trabalho .....</b>	<b>3</b>
<b>2.</b>	<b>Requisitos .....</b>	<b>4</b>
<b>2.1.</b>	<b>Requisitos Funcionais .....</b>	<b>4</b>
<b>2.2.</b>	<b>Requisitos Não-Funcionais .....</b>	<b>4</b>
<b>2.3.</b>	<b>Restrições Arquiteturais .....</b>	<b>4</b>
<b>3.</b>	<b>Modelagem e projeto arquitetural .....</b>	<b>5</b>
<b>3.1.</b>	<b>Modelo de casos de uso .....</b>	<b>5</b>
<b>3.1.1.</b>	<b>Detalhamento dos casos de uso .....</b>	<b>6</b>
<b>3.2.</b>	<b>Modelo de componentes .....</b>	<b>11</b>
<b>3.3.</b>	<b>Modelo de implantação .....</b>	<b>12</b>
<b>3.4.</b>	<b>Modelo de dados .....</b>	<b>13</b>
<b>3.5.</b>	<b>Implementação .....</b>	<b>14</b>
<b>3.6.</b>	<b>Implantação .....</b>	<b>14</b>
<b>4.</b>	<b>Link GitHub .....</b>	<b>15</b>

## **1. Apresentação**

Este documento tem por finalidade descrever a arquitetura de uma aplicação web desenvolvida para a ASSOCIAÇÃO DE CEGOS LOUIS BRAILLE - “Lar das Cegas”, e tem por principal objetivo auxiliar na divulgação e captação de doações e recursos para a realização dos eventos acontecidos em prol da associação.

### **1.1. Problema**

O Lar das Cegas é uma associação que abriga mulheres com deficiência visual com a intenção de ajudá-las a conquistarem autonomia. Para que possa manter o seu funcionamento, eventos são realizados para a captação de recursos. Entretanto, eles não são divulgados de forma muito efetiva. Devido a isso, recebem um público pequeno, sendo em sua maioria pessoas que já conhecem a associação e acompanham sua página no Facebook.

### **1.2. Objetivos do trabalho**

O objetivo geral deste documento é descrever o projeto arquitetural da aplicação, que será um totalmente web.

A aplicação será desenvolvida para que haja uma maior divulgação dos eventos realizados pela associação. Além disso, ela poderá ajudar os organizadores a terem uma noção da quantidade de pessoas que estarão presentes. Desse modo, poderão se organizar para uma melhor utilização dos recursos. Também será possível que pessoas interessadas façam doações para ajudá-la.

## **2. Requisitos**

Com o intuito de disponibilizar de forma mais fácil informações referentes a associação em relação aos seus eventos, a aplicação a ser desenvolvida possui alguns tópicos que devem ser seguidos. Estes tópicos são definidos como requisitos, que são separados em dois tipos.

## **2.1. Requisitos Funcionais**

São requisitos funcionais que serão aplicados no desenvolvimento desta aplicação:

1. Deve possuir Implementação de um login. O usuário com os campos CPF e senha, e estes devem ser validados.
2. Permitir ao administrador consultar aos eventos da associação, de forma consolidada, com os seguintes filtros: nome, data ou local.
3. Permitir ao usuário visualizar um evento e caso deseje realizar uma doação.
4. Cadastro
  - O administrador deve cadastrar o evento com os seguintes campos: Nome do evento, imagem, data e descrição.

## **2.2. Requisitos Não-Funcionais**

Os requisitos Não-Funcionais são:

1. A aplicação deve possuir uma interface responsiva.
2. O sistema suportará até 500 usuários simultâneos.
3. O sistema deve responder em até 4 segundos em situações normais de conectividade.

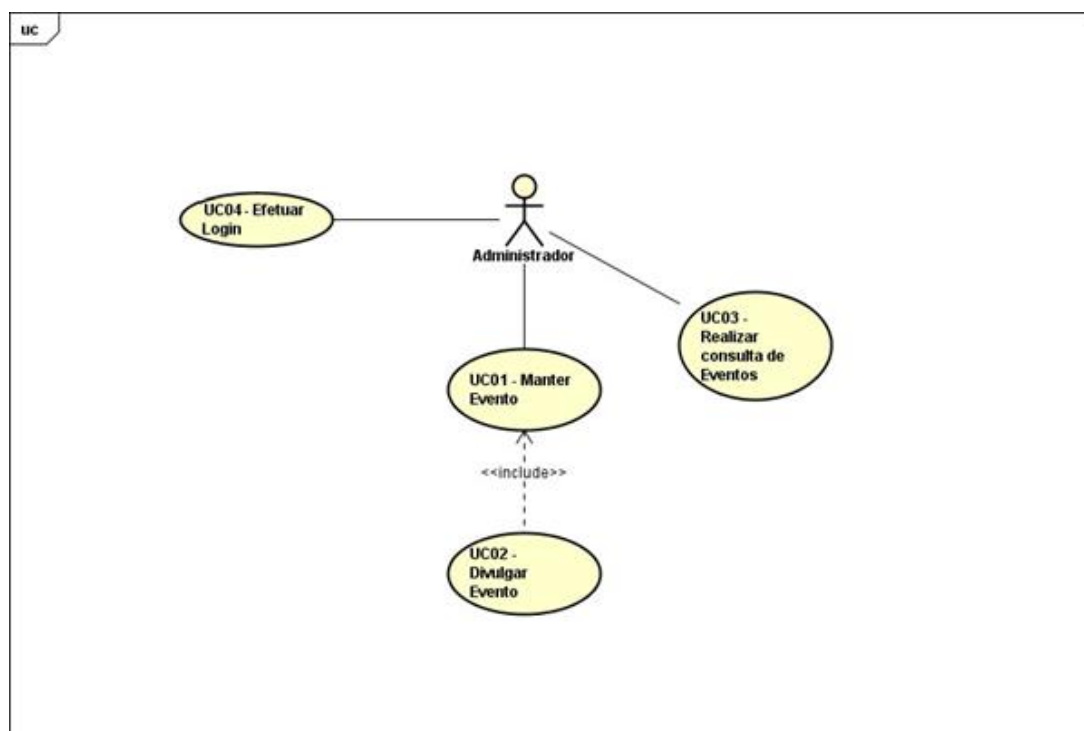
## 2.3. Restrições Arquiteturais

1. Possuir arquitetura em camadas, com diversas possibilidades de acesso: smartphone, tablet, web, desktop;
2. Ser construído com uma parte de front-end e outra de back-end, que irão se integrar de maneira transparente (heterogeneidade interna X transparência);
3. Priorizar os seguintes requisitos arquiteturais do desenvolvimento portabilidade, manutenibilidade, escalabilidade;
4. Ser desenvolvido segundo as boas práticas de arquitetura de software;
5. Seguir as premissas da orientação a objetos na análise e no projeto;
6. Utilizar linguagem PHP integrado com banco de dados Mysql para o back-end e o front-end será utilizado HTML, CSS e Bootstrap.

## 3. Modelagem e projeto arquitetural

Esta seção aborda o detalhamento arquitetural da solução proposta.

### 3.1. Modelo de casos de uso



Fonte: Elaborada pelo autor

### 3.1.1. Detalhamento dos casos de uso

Classificação	Caso de Uso	Justificativa
<b>Alto</b>	UC01 - Manter Evento	Afeta a tabela principal do banco de dados.
<b>Alto</b>	UC02 - Divulgar Evento	O conteúdo principal do aplicativo, falhas devem ser mínimas.
<b>Médio</b>	UC03 - Realizar Consulta de Evento	Exige acesso a todos os eventos registrados no sistema.
<b>Baixo</b>	UC04 - Visualizar Eventos	Efeito mínimo na arquitetura.
<b>Alto</b>	UC05 - Efetuar Login	Utiliza os métodos e classes de segurança do sistema.

#### 3.1.1.1 Caso de uso: UC01 - Manter Evento Sumário:

Seção onde o administrador pode criar, alterar e apagar os eventos da associação.

##### Ator primário:

Administrador.

##### Pré-condições:

O administrador precisa estar logado no sistema.

##### Fluxo principal:

1. O caso de uso UC01 - Manter Evento inicia-se quando o administrador, já está presente na tela principal e deseja criar, alterar ou apagar um evento do sistema.
2. O administrador seleciona uma das opções do caso de uso, que são criar um novo evento, alterar um evento ou apagar um evento.

< FA 001 - Novo Evento >

< FA 002 - Alterar Evento >

< FA 003 - Apagar Evento >

< FA 004 - Cancelar Ação >

3. O sistema valida a ação do administrador.

<FE 001 - Tratamento de Erro >

4. O sistema emite uma mensagem avisando que a ação foi realizada com sucesso.
5. O caso de uso é encerrado com sucesso.

**Fluxo alternativo:**

< FA 001 - Novo Evento >

1. O sistema exibe um formulário a ser preenchido.
2. O administrador preenche o formulário.
3. O administrador aperta o Botão “Prosseguir”.
4. O fluxo retorna ao passo 3 do fluxo principal.

< FA 002 - Alterar Evento >

1. O administrador seleciona no datagridview qual evento deseja alterar.
2. O sistema exibe um formulário com todos os campos preenchidos referentes ao evento selecionado.
3. O administrador realiza as alterações necessárias.
4. O administrador aperta o Botão “Prosseguir”.
5. O fluxo retorna ao passo 3 do fluxo principal.

< FA 002 - Apagar Evento >

1. O administrador seleciona no datagridview qual evento deseja apagar.
2. O sistema exibe um formulário com todos os campos preenchidos referentes ao evento selecionado. Esses campos não podem ser alterados.

3. O administrador confere se o evento selecionado é o que ele deseja apagar.
4. O administrador aperta o Botão “Prosseguir”.
5. O fluxo retorna ao passo 3 do fluxo principal.

< FA 004 - Cancelar Ação >

1. O administrador cancela a ação solicitada.
2. O estado geral do sistema permanece inalterado.
3. O caso de uso é encerrado.

**Fluxo de exceção:**

<FE 001 - Tratamento de Erro >

1. O Sistema identifica o erro.
2. O Sistema emite um alerta ao administrador avisando do erro encontrado e as suas possíveis causas e soluções.
3. O sistema redireciona o administrador ao passo anterior do erro encontrado.

**Pós-condições:**

As ações realizadas pelo administrador são registradas no banco de dados e salvas em um log de dados.

### **3.1.1.2 Caso de uso: UC02 - Divulgar Evento**

Sumário: Após o cadastro de um evento, o mesmo será divulgado para os usuários logo em seguida através do método DivulgarEvento.

**Ator primário:**

Sistema

**Pré-condições:**



O caso de uso UC01 - Cadastrar Evento tem todos os seus passos no fluxo principal completos.

**Fluxo principal:**

1. O sistema irá liberar para os usuários visualizarem o evento.
2. O caso de uso UC02 - Divulgar Evento é encerrado.

**Fluxo alternativo:**

Não há fluxo alternativo para esse caso de uso.

**Fluxo de exceção:**

Não há fluxo de exceção para esse caso de uso.

**Pós-condições:**

O sistema emite uma mensagem, avisando ao administrador que o evento foi divulgado.

### **3.1.1.3 Caso de uso: UC03 - Realizar Consulta de Evento:**

Seção onde o administrador tem acesso às informações de todos os eventos cadastrados, através de uma ferramenta de pesquisa.

**Ator primário:**

Administrador

**Pré-condições:**

O administrador precisa estar logado no sistema e para efetuar uma busca é necessário ao menos um evento cadastrado.

**Fluxo principal:**

1. O caso de uso inicia quando o administrador deseja iniciar uma consulta dos eventos cadastrados no sistema.
2. O administrador seleciona a função de consulta de dados dos eventos.

3. O administrador preenche o campo de busca com as informações necessárias (Nome do evento, data ou local).
4. O sistema automaticamente filtra através das informações do campo de busca.
5. O sistema exibe os eventos encontrados.
6. O caso de uso é encerrado com sucesso.

**Fluxo alternativo:**

Não há fluxo alternativo para esse caso de uso.

**Fluxo de exceção:**

Não há fluxo de exceção para esse caso de uso.

**Pós-condições:**

O sistema retorna os eventos referentes aos eventos procurados.

**3.1.1.6 Caso de uso: UC08 - Efetuar Login:**

Seção em que é feita a verificação e atribuição do acesso ou não do portador de determinada credencial.

**Ator primário:**

Administrador.

**Pré-condições:**

O administrador deve ter o seu cadastro presente no sistema sendo isto verificado na tentativa de login.

**Fluxo principal:**

1. Esse caso de uso inicia quando o administrador ou usuário tenta efetuar o login.
2. O administrador seleciona a função de login.
3. O administrador informa ao sistema suas credenciais.

4. O administrador pressiona (Clica) o botão “Entrar”.
5. Após o clique, o sistema verifica se os dados inseridos são válidos e garante o acesso ao administrador ou usuário.

< FE001 - Dados Inválidos >

6. O caso de uso é encerrado com sucesso.

**Fluxo alternativo:**

Não há fluxo alternativo para esse caso de uso.

**Fluxo de exceção:**

< FE001 - Dados Inválidos >

1. O sistema exibe um aviso informando que dados inválidos ou não cadastrados foram inseridos no formulário de login.
2. O sistema volta ao passo 3 do fluxo principal.

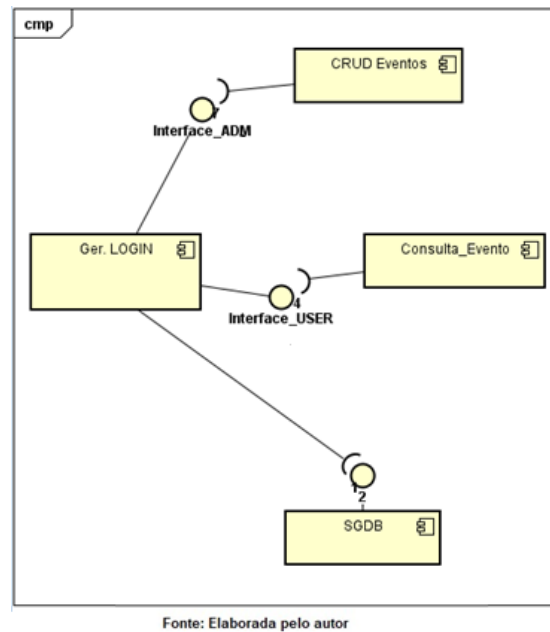
**Pós-condições:**

O administrador ganha acesso à área restrita referente a sua conta no sistema.

### **3.2. Modelo de componentes**

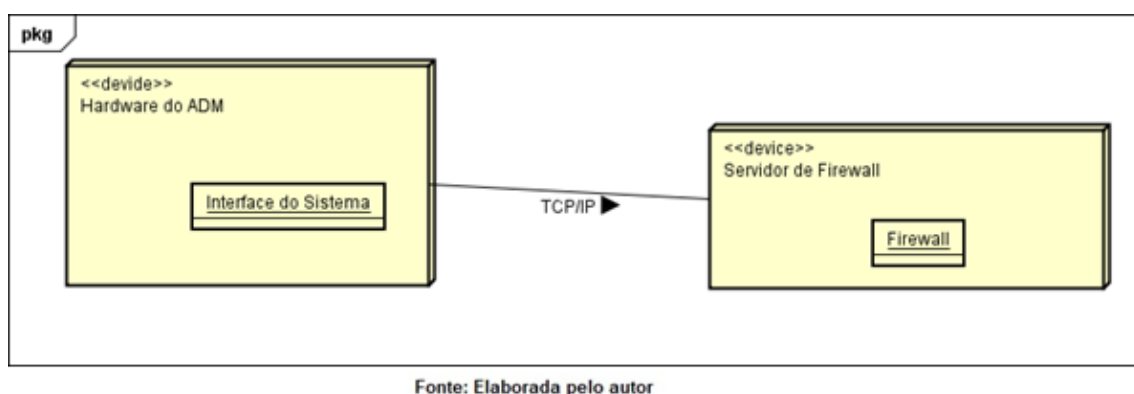
Tendo em vista que está será uma aplicação web (linguagem PHP), seguem os principais componentes envolvidos nesta aplicação:

- SGBD: Componente de banco de dados em que utilizaremos o Mysql para armazenamento.
- Gerenciamento de login: ADM (administrador): com seus componentes de CRUD de eventos (inserir, editar, alterar, excluir) e emissão de relatórios de arrecadação e presença no evento.



### 3.3. Modelo de implantação

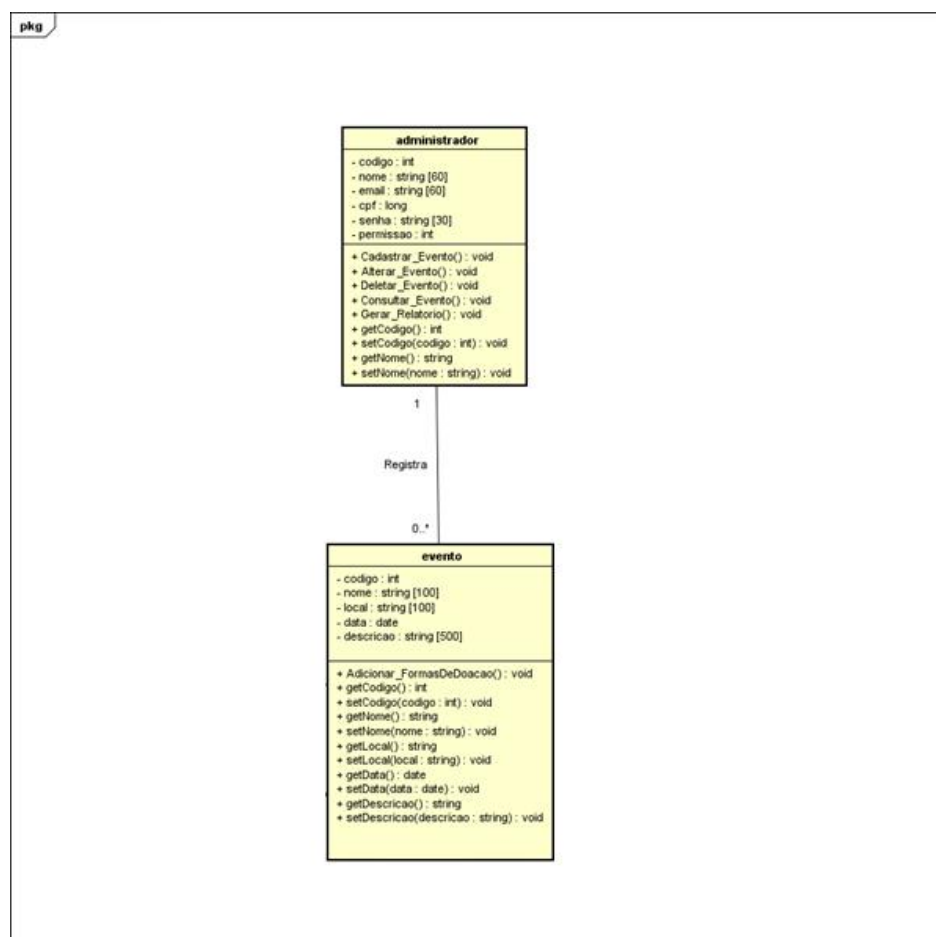
Partindo do pressuposto de que já existe uma rede banda larga de internet wifi em funcionamento, assim como todas as configurações e instalações já realizadas, segue abaixo um diagrama de implantação da aplicação descrevendo apenas a parte de backend (salientando de o front-end serão dispositivos móveis).



Configuração mínima recomendada:

- Quantidade de máquinas: 01
- Sistema operacional: Win 10 ou linux,
- Memória ram: 3gb
- HD: 250gb
- Processador: Quadcore 1.7 Ghertz ou superior
- Placa de rede wireless devidamente configurado

### 3.4. Modelo de dados



Fonte: Elaborado pelo autor

### **3.5. Implementação**

A implementação do site, começa desde a construção do front-end com HTML, CSS e Bootstrap e o back-end PHP com a biblioteca PDO. O bando de dados será armazenado pela plataforma Mysql. O conteúdo do site, será repassado pela ONG e os eventos serão cadastrados e publicados pelo administrador.

### **3.6. Implantação**

A implantação será no host da Umbler, com o controle do GitHub. O domínio e email serão comprados e controlados pela ONG.

## **4 Link do GitHub**

<https://github.com/idavifelipe/lardascegas-tw.com.br>